UNIVERSITI MALAYA

SARJANA MUDA SAINS KOMPUTER DENGAN KEPUJIAN

PESAT APPLICATION

Personal Security Administrator Tool

LAW YONG JIN

WEK010125

(SESSION 2003/2004)

PROJEK ILMIAH TAHAP AKHIR

Under supervision of

MRS. FAZIDAH BINTI OTHMAN

Moderated by

En Nor Badrul Anuar Juma'at

This project is submitted to the Faculty of Computer Science and Information Technology, University of Malaya in partial fulfillment of the requirement for the Degree of Bachelor of Computer Science

ABSTRACT

The thesis report is written to descript the development of PESAT application in depth. It contains from the project objective, definition, scope from the beginning; extends to literature review, study on existing application, software process model, and application development tool, functional and non-functional requirements; and ended with the system structure design, system process design and user-interface design. The report will be furthered with more details on system implementation, testing and evaluation on the coming semester. It is well-structured within layouts and contents to cover as much information as an application development proposal.

PESAT is a security and administration application that tends to protect the security of personal computer with the administration of the computer on the same time. PESAT enhances the current existing security and administration tools in the market, by improving the weaknesses and expanding the strengths of those existing application. In short, PESAT is a well-organized, securable, exceptional and superb security administrator tool that is going to be developed. It is built using the current application development tools with a resourceful and fundamental research on the security and administration arenas.

The high expectation towards the outcome of PESAT application is to construct an application that approaches the conceptual of computer security as well as administration. In order to step in a successive development of PESAT, it must be well-planned, well-structured and well-organized to meet every functional and non-functional requirement.

ACKNOWLEDGEMENT

Upon the development of PESAT application, many people have been pouring a quantitative of helpful assistances, supportive encouragements, and priceless advices on several aspects regards to the PESAT development. Without their useful, handy, constructive and practical assistance, the PESAT application development might meet with many difficulties.

Consequently, I wish to dedicate my sincere gratifier and string of appreciation to the follow person:

1. Puan Fazidah binti Othman, as my supervisor who takes full dedication to the PESAT application, giving valuable suggestions and guidelines to a successive PESAT application development.
2. Mr. Nor Badrul Anuar Juma'at, as my modulator who willing to spend valuable times with my PESAT application, commenting beneficial ideas on PESAT application.
3. Mr. Ang Tan Fong, as my personal academic advisor who willing to assist me in solving numerous difficulties that comes along in the PESAT application development.
4. My family and fellow of friends, who give full supportive and passionate encouragements along with the PESAT application development.

TABLE OF CONTENT

Report Outline:

Chapter Description:

Chapter 1: Introduction

Chapter 2: Literature Review

Chapter 5: System Design

Chapter 6: System Implementation

Chapter 7: System Testing

Chapter 8: System Evaluation And Conclusion

LIST OF TABLE

LIST OF FIGURE

CHAPTER 1
INTRODUCTION

1.1 Introduction

The introduction chapter of the thesis project report will deals in detail about the project overview, its objectives, scope and definition. Besides, the identified problems, the project significant and expectations are listed in this chapter as a basis and fundamental in the development of the thesis project.

The rationale of this chapter is to initiate and develop the thesis project inline with its objectives and scopes so that it will not bias from its original development purposes.

1.2 Project Overview

Security administration can be overseen as a most vital part in any computer system. Without the sufficient security measurements, any sensitive information that supposed to be confidential will easily flow out to the public. The current software market trend reviews that although there is a vast array of computer application software, but software relates to the security administration program is still a lack in the market.

Consequently, 'Personal Security Administrator Tool' or acronymic as PESAT is developed to fit the market need. It is a simple yet powerful tool that can be widely

deployed in the most contemporary personal computer, administrating the security measurements and tweaking as well as shielding the entire computer system to the highest point of the security level.

1.3 Project Definition

Personal Security Administrator Tool is a complete personal computer security program that will extends the functionality of administration in Windows XP operating system. PESAT is packaged with a delight, genuine, elegant and user-friendly interface so that user can drive the application easily.

Conceptually, user that deploys the PESAT application will be acting as an administrator, who has the right to access all of the security and administration options and configure the functionalities as provided in the PESAT application. The application is protected by a strong password, thus the other user who uses the same computer can not simply access the program, and he or she will be restricted by the security level set by the administrator. In such, the administrator can use the PESAT application to apply numerous security administration functions in the computer within the mouse click. The PESAT application is running in background and an iconic icon will be placing in the system tray for easy access.

In deep analysis, PESAT is written mostly using the Microsoft Visual Basic 6 and Windows Script Host, WSH programming language. Visual Basic language is used to program most of the security function provided. Acting as Windows administrator tool, PESAT is often relate to the kernel and core of the operating system. Consequently, WSH is deployed in the PESAT development. WSH creates an environment for hosting scripts. It is language-independent for WSH-compliant scripting engines for the Windows Operating Systems.

PESAT is an application that provides a lot of security functions as well as administration tweakings regard to a personal computer. It is said that in Windows XP Operating System, there are many hiding functions unknown to the users, like Windows

Auto-login, and hiding C-Drive. Consequently, PESAT application is developed to uncover some of the hiding functionalities. It is independent software that is used together with Windows XP. Windows XP is chosen as the application platform because in the Windows family operating system series, Windows XP has emerged as the most securable operating system, compare to Windows 2000 which is getting outdated in this day while Windows 98 and Windows ME are the old brand operating system which is said to be fallen into desuetude.

Dealing with the functionality, PESAT have the ability to manage a number of the security in the computer which uses the Windows XP. Firstly, PESAT has a password authentication interface to log into the application. Besides, a password generator is provided to assist the user to generate a securable and strong password to assure that no occurrence of any illegal access to the PESAT application. The PESAT application has the ability to notify the user of the virus scanner update, or it can be set to automatically download the virus definition periodically. This function can only be used with the Norton Antivirus 2003 Software.

Next, the PESAT application can lock any file, folder or even program. Only the user with the correct password can unlock the file, folder or program. This is to protect the sensitive data in the computer. It has an interface so-called Program Aliases Management to locate or relocate the actual path of the installed program. It also can monitor the Add or Remove Program, by listing the installed program and interface to add or remove the program within the computer. PESAT have an interface that links to the Control Panel, which means that any of the computer setting can be made without the PESAT application. It also can check the currently services Windows is running and the function whether to stop any of the services if needed by the user. PESAT has the features that can one click to Logoff, Shutdown or Restart the computer. This is indeed a convenient way compare to the typical computer shutdown method.

In term of security functions, PESAT provides a vast array of security options to be applied in the computer system. It can totally disable the Control Panel, or partially hide

some of the applet of the Control Panel in the computer system. This is to disallow changes or modification by any unauthorized user to the computer system. PESAT can also hide the drive visibility, for instance, hiding the floppy drive or entire hard-drive-C. It can also lock My Computer, My Documents and Recycle Bin in the desktop, or totally lock the whole desktop property; provides screen saver password protection; disable some of the command in the Start Menu like RUN, FIND and etc; disable the auto run feature of the CD-Rom; disable the Registry Editor in the computer; reconfigure the menu show delay in the Start Menu; Control the mouse right-click menu, whether to totally disable or modify the menu; and lock the communication ports, COM port and printer port, LPT port. Dealing with the Internet security, the PESAT application can delete the internet cookies, internet temporary files, internet histories, Internet Explorer typed history and recent document history.

In short, PESAT is a handy security tool which provides a lot of simplicity to control the security issue in the computer. The PESAT application definitely will enhance the security in the Windows XP operating system.

1.4 Project Objective

In term of a successive development of the PESAT application, a set of the objectives has been clearly defined. The objectives is a vital part of the project development as it will inevitably act like a guide to lead in the correct path and practice measurements that must be taken and considered in the PESAT application, in order to meet the requirements and goals of the project. Below are the listings of the PESAT development objectives:

1. To provide a more securable, applicable, credential, scalable and reliable way to protect and administrate the computer, using an elegant, classical, user-friendly and non-technical interface which is convenient to the user to apply security measurement to their computer system
2. To protect the important information or sensitive data reside in the computer system from being accessed by the unauthorized intruders

3. To protect and secure the computer system from being infected by the new-borne-viruses using existing antivirus scanner software
4. To provide a convenience way of easy access to the security and administration setting in the Windows XP Operating System and enhance the right of administrator, which can restrict and apply immense of exceptional and outstanding security measurements to other users, as well as to prevent users from changing or altering the setting within the computer system
5. To be developed into a small-scale-package application that is able to monitor the computer system in background without consuming a lot of memory resources

1.5 Project Scope

The PESAT application is aimed to provide a vast array of security measurements as well as the administration measurements to the user to protect their computer system.

PESAT is innovated to maneuver over the Windows XP Professional operating system environment. It does not work in operating system environment other than Windows XP as most of the functionalities are written based on the Windows XP system files. Some functionality might not work properly in operating system other than Windows XP.

Besides, it is developed for the personal deployment only. 'Personal' in term of the PESAT does carry the meaning of only one user profile exists in the computer. PESAT application is not specially designed for the networking environment, where more user profiles exist in one computer. In multi-programming operating system operating system likes Windows XP, PESAT application suits only for one user profile.

Some of the tweakings might take effect only after the computer has reboot.

On basis, it can be categorized into several major sections to be developed:

PESAT Application Security

1. A password protected interface to open the application.
2. A password generator to assist the user to build a strong password.
3. Selective application hide/show in system tray.
4. Selective application auto start or manual start.

Permission and Authorization Security

1. A File Locker to provide password-protected file.
2. A Folder Locker to provide password-protected folder.
3. An Application Restriction Management to restrict or allow the application to be running in the system.
4. A Program Aliases Management to match/locate/relocate the correct path to the application installed in the computer.
5. An Add/Remove Program to simplify the task to add program into the system or remove the installed program within the system.

Computer Tweaking Security

1. A General Computer Management to disable control panel, or partially control panel applet; disable registry editor, REGEDIT from the user; disable CD-Rom auto run feature; a link to control panel easy-access; remove RUN, FIND, SHUTDOWN and to alter menu show delay duration in the START MENU; control the right mouse click function.
2. A Drive Visibility Management to hide or show drives.
3. A Port Visibility Management to hide or show communication port, COM port or printer port, LPT port.
4. A Desktop Management to lock the desktop and password-protected screen saver.
5. An Internet Management for user to delete cookies, temporary files, history, IE-typed history and recent document histories.

Anti-Virus Definition Security

It is developed with the integration in Symantec Norton Antivirus 2003 only.

1. An Update Notification Alert to prompt user about the last date virus definition being updated and whether new patch is being released from the antivirus vendor.
2. An Update Automation Tool to provide auto or manual virus definition update.

Windows Service and Power Management Security

1. One Click Button Windows termination Shutdown, Log-off and Restart.
2. Windows Services Security.
3. A Services Checker to monitor services running in the system.
4. A Services Stopper to halt the unwanted services.

1.6 Project Motivation

There is a quantitative of relevant security and administration applications in the current software market. From the conducted researches on the most of the application, a lot of the weaknesses do exist leading many security flaws and holes to the intruders. On the other hand, the administration in the Windows XP Professional Operating System might be a little fuzzy and inconvenience to most of the computer users, especially the new PC users.

In short, the development of PESAT application is motivated by the limitations of the existing security and administration applications. The motivations are listed as below:

1. The current computer system is being threatened by a momentous of virus attack incident. Consequently, the virus scanner element is a must in the security application. Existing security application reveals that most of them do not consider the virus scanner key element to be adhered in their application. This draws to the PESAT application to develop and incorporate a virus scanner into a compact security application that is able to update the virus definition regularly and automatically. This measurement is vital in a computer system as it keeps the virus definition in the virus scanner up-to-date and minimizes the threat of the newest

virus explosion to the lowest point. PESAT application is using Norton Antivirus 2003 as the virus scanner.

2. The existing security application does not provide sufficient authentication to the application. This remarks that any of the users could alter the security measurements within the application. Likewise, the PESAT application is developed using an authentication conceptual. Only the administrator to the PESAT application can tweak, alter or apply security measurement to the computer system. PESAT application is using strongly encrypted password to authenticate the application log-in.

3. Tweaking the administration within the Windows XP Operating System might be a daunting task to mostly users. The PESAT application is an integrated application providing both security measurements and administration tweaking equally. It simplifies the Windows administration tweaking in a convenient way. Besides, PESAT application appends a security administration to the operating system as all the administration tweaking can only be completed through the application.

1.7 Project Limitation

The development of the PESAT application is limited by some of the factors. The limitations are listed as below:

1. The PESAT application is expected to be running at the windows background, so that it can monitor the windows security time by time. This denotes that the application must be packaged in a miniaturized-size, so that it will not waste the windows resources.

2. Relate to the Virus Scanner Security, the PESAT application is expected to well-communicate with the antivirus software. In such, the difficulty is to find appropriate antivirus software that can be integrated in the PESAT application. Addition to this, different antivirus would have the different core files, which means that PESAT should recognize various vendor and version of the antivirus software. In the PESAT

application, Symantec Norton Antivirus 2003 antivirus has been elected for the demonstration of function of Virus Scanner Security.

3. The PESAT application is developed mainly to be deployed in the Windows XP operating system platform environment. It is not fully functional in other Windows like Windows 98, ME or 2000. This limitation is inevitable because different operating system contains different system files. In addition, Windows Script Host in the PESAT application is only fit in the Windows XP environment.

1.8 Project Significant

1. This thesis application is significant in the sense of provides an alternative security tool that can be widely deployed in the Windows XP operating system platform. The wide range of the security options in the PESAT application impress itself as a solution regards to the security issue in the computer.

2. The PESAT application is targeting for the personal computer users, which normally oversee the computer security issue as an insignificant task because of the complicated steps to apply the security settings. Thus, the outcome of the thesis project, PESAT application will be the ultimate yet handy tool to apply mostly security measurements in their personal computer in a convenience way.

3. Besides, PESAT is believed to be dominant and significant in the computer-based business environment such as Cyber-cafe and school computer. In those places, PESAT plays a role to prevent all unauthorized changes to the computer system. The user can only perform basic function of the computer, like processing a document, playing game or serving net; not to alter the computer settings.

4. The most significant part of the PESAT application is to provide a shield to the important files and folders. In sense of this, all of the sensitive data would not be accessed to the public as the PESAT application protects the files or folders with the

mask password. In such, the PESAT application is pushing the security issues in the computer to the highest level.

1.9 Project Expectation

The PESAT application is developed in aim to provide a wide range of security choices for the user to be apply in their computer, purposely to protect their computer system, especially the sensitive information, from being accessed by the intruder. PESAT is expected to cover as much as possible security measures in its implementation and overcome the weaknesses of the similar security applications exists in the software market. This application is expected to introduce a more creative, innovative, informative, interactive and securable application that will integrating in the Windows XP operating system.

Besides, the application is tending to draw the attention of the typical computer users of their computer security issues. PESAT denotes how vulnerabilities of a computer system in term of security, especially in the Windows base operating system. With the PESAT in use, the security leakages in the operating system environment tend to be reduced into the minimum condition.

Finally, the output of PESAT application is expected to be a small scale application, not using many system resources, maximum security measures, excellent designated user interface, not using many computer jargon and ease to use with the Windows XP operating system environment.

1.10 Project Schedule

A project schedule is essential for the management of time so that all the schedule tasks can be accomplished and completed within the period of time. In such, a project schedule is well planned to ensure that all the project development process in the PESAT development will works out smoothly.

Key Activities	July 2003				August 2003				November 2003				December 2003				January 2004			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Literature Review																				
System Analysis																				
System Design																				
Coding																				
Unit Testing																				
Integration Testing																				
System Testing																				
Documentation																				

Table 1.1: PESAT Application Project Schedule

1.11 Summary

In the era of inter networking, work groups and digital highways, it becomes more and more important to protect personal sensitive data from malicious or unintended harm. More and more documents and information are migrated to the computer in the form of files stored in the hard disk. More and more computers are linked together, exposing those files to dangers of being destroyed, examined, or adulterated intentionally or not by other users (hackers) of the network.

In such, the PESAT application is designed to help computer user to protect the system against such intrusions. It is an application specially designed to work as an administrator to extend many security measures to the computer system.

This chapter tends to draws the application in general. It clearly states the project overview, definition, objectives, scope, motivation, significant, expectation and schedule. Literature review is following up this chapter.

CHAPTER 2
LITERATURE REVIEW

2.1 Introduction to Literature Review

PESAT is acronymic to Personal Security Administrator Tool. This is an innovative, inventive yet progressive application developed to provide many security measures in term of computer administration for the need of the Windows XP operating system environment. The development of the PESAT application extends within several areas of computer administration and security knowledge. Consequently, literature review process has been carried out in order to collect more informative resources, ingenious ideas and superior technologies so that the development of the PESAT application would be the absolute security administration application overcoming the weaknesses of the existing security tools as well as extending their strengths in the software market. Research on current existing security administration tools is analyzed base on their strengths and weaknesses; application development tools is discussed in depth to provide

a fundamental basis on what the technologies that are going to be used in the system analysis process later.

2.2 Background Study

The PESAT application development is manipulating the conceptual idea of security in the computer with the administration purpose. So in the literature review section, a resourceful of information regards to computer security and administration areas are adhere.

2.2.1 Computer Security Issue

Computer Security

The purpose of computer security is to devise ways to prevent the weaknesses from being exploited. A system is to say as 'secure' if only every preventive measures have made the most sense. Three important aspects to be related to the computer security are confidentiality, integrity and availability. Computer items must be protected only until they lose their value. They must be protected to a degree consistent with their value. [8]

There is information on the computer that is worth protecting. Everybody has a right to privacy, and most computers and computer software in use today isn't designed with this in mind. Many people wrongly assume that it is easy to prevent unauthorized access to data stored on a PC and fail to realize how easily a person with the correct knowledge can access their most private of files. [9][10]

Threat and Vulnerability

Vulnerability is a weakness in the security system. It applies within three broad categories of system resources, namely hardware, software and data. It prevents from reaching one or more of the three security goals. For instance, a particular system may

be vulnerable to unauthorized data manipulation because the system does not verify a user's identity before allowing data access.

A threat to a computing system is a set of circumstances that has the potential to cause loss or harm. There are many threats to a computer system, including human-initiated and computer initiated ones.

Interception, Interruption, Modification and Fabrication

There are four classes of threats, namely Interception, Interruption, Modification and Fabrication, faced by the personal computer today. *Interception* means that some unauthorized party has gained access to an asset. The outside party can be a person, a program or a computing system. *Interruption* is an asset of the system becomes lost, unavailable or unusable. If an unauthorized party not only access but tampers with an asset, the threat is a *modification*. An unauthorized party might create a *fabrication* of counterfeit objects on a computing system. [4]

General Security Principle

There are many general security principles which should be familiar with; one good place for general information on information security is the Information Assurance Technical Framework (IATF). Often computer security objectives or goals are described in terms of three overall objectives:

1. Confidentiality, also known as secrecy, meaning that the computing system's assets can be read only by authorized parties.
2. Integrity, meaning that the assets can only be modified or deleted by authorized parties in authorized ways.
3. Availability, meaning that the assets are accessible to the authorized parties in a timely manner, or as determined by the systems requirements. The failure to meet this goal is called a denial of service.

Security in computing addresses these three goals. One of the challenges in building a secure system is finding the right balance among these goals, which often conflicts. But balance is not all, where these three characteristics can be independently, can overlap or even be mutually exclusive. Some people define additional major security objectives, while others lump those additional goals as special cases of these three.

Privacy is sometimes addressed separately from confidentiality; some define this as protecting the confidentiality of a user instead of the data. Most objectives require identification and authentication, which is sometimes listed as a separate objective. Often auditing, also called accountability is identified as a desirable security objective. Sometimes ``access control" and ``authenticity" are listed separately as well.

A good general security principle is ``defense in depth"; it should have numerous defense mechanisms in place, designed so that an attacker has to defeat multiple mechanisms to perform a successful attack.

Software Engineering Security Principle

Least Privilege

Each user and program should operate using the fewest privileges possible. This principle limits the damage from an accident, error, or attack. It also reduces the number of potential interactions among privileged programs, so unintentional, unwanted, or improper uses of privilege are less likely to occur.

Economy of Mechanism/Simplicity

The protection system's design should be simple and small as possible.

Open Design

The protection mechanism must not depend on attacker ignorance. Instead, the mechanism should be public, depending on the secrecy of relatively few items like passwords or private keys. An open design makes extensive public scrutiny possible,

and it also makes it possible for users to convince themselves that the system about to be used is adequate.

Complete Mediation

Every access attempt must be checked; position the mechanism so it cannot be subverted.

Fail-safe Default

The default should be denial of service, and the protection scheme should then identify conditions under which access is permitted.

Separation of Privilege

Ideally, access to objects should depend on more than one condition, so that defeating one protection system won't enable complete access.

Least Common Mechanism

Minimize the amount and use of shared mechanisms (e.g. use of the /tmp or /var/tmp directories). Shared objects provide potentially dangerous channels for information flow and unintended interactions.

Psychological Acceptability / Easy to Use

The human interface must be designed for ease of use so users will routinely and automatically use the protection mechanisms correctly. Mistakes will be reduced if the security mechanisms closely match the user's mental image of his or her protection goals.

Security Application Classification

Many different types of applications may need to be secured completely, whether for the administration use or protect some sort of sensitive information. Some common types are listed as below.

Application as Viewer of Remote Data

Programs used as viewers (such as word processors or file format viewers) are often asked to view data sent remotely by an un-trusted user (this request may be automatically invoked by a web browser). Clearly, the un-trusted user's input should not be allowed to cause the application to run arbitrary programs. It's usually unwise to support initialization macros.

Application used by the Root Administrator

Such programs shouldn't trust information that can be controlled by non-administrators.

Local Server or Daemon

Network-accessible servers (sometimes called network daemons).

Web-Base Application

These are a special case of network-accessible servers, but they're so common they deserve their own category. Such programs are invoked indirectly via a web server, which filters out some attacks but nevertheless leaves many attacks that must be withstood.

Applet

There are several security viewpoints here; the implementer of the applet infrastructure on the client side has to make sure that the only operations allowed are ``safe'' ones, and the writer of an applet has to deal with the problem of hostile hosts.

'Setuid/setgid' Program

These programs are invoked by a local user and, when executed, are immediately granted the privileges of the program's owner and/or owner's group. In many ways these are the hardest programs to secure, because so many of their inputs are under the control of the un-trusted user and some of those inputs are not obvious.

Security Countermeasure in Depth

Harm occurs when a threat is realized against vulnerability. To protect against harm, user must neutralize the threat, close the vulnerability, prevent by blocking the attack, deter by making the attack harder, deflect by making another target more attractive, and recover from its effect. Below are some of the security countermeasures.

1. Using encryption approach to virtually nullify the value the value of an interception and possibility of effective modification or fabrication.
2. Internal program control by enforcing security restrictions, such as access limitations in a database management program.
3. Operating system and network system controls by enforcing limitations to protect each user from all other users.
4. Using independent control programs like password checkers, intrusion detection utilities or virus scanners to protect certain vulnerability.
5. Using hardware devices that have been created to assist in providing computer security, for instance devices to verify users' identities and hardware implementations of encryption.
6. Practice physical control instead of seeking more sophisticated approaches.

Controls against vulnerabilities must be used, and used properly, to be effective; they must be efficient, easy to use and appropriate.

Security Configuration in Depth

Hardware

The first thing to consider with the security of the sensitive data is whether people can access the computer in physically or not. With a home personal computer, it is more likely to address the risk to come from outside, a thief maybe, so think about securing the room within the computer is locating. Below are some general guidelines to protect the computer in term of hardware.

1. Keep the personal computer in a locked steel case bolted to the floor or the wall, make sure the case can't be opened or drives removed from the front.
2. If the case has a door on the front, always keep it locked.
3. If the case has a lock that disables the keyboard, always use it when the machine is switched off or left unattended.
4. If there is no need to use a floppy drive, remove it. Alternatively consider purchasing a good floppy drive lock. Beware though, many floppy drive locks share the same key or are very easy to pick.
5. Keep all keys in a safe place.

Firmware

The system BIOS is the first program that runs when starting the computer. It is a preventive measurement if the computer is starting up by using a BIOS password. By then, it will deter the casual attacker and slow down a more determined attack. User should be aware though that the BIOS password isn't a total solution even if it seems like it, it's easy enough to open the case and set a jumper on the motherboard to reset the BIOS to its factory defaults, thus removing the password. It's also similarly easy to determine the BIOS password if user has access to the machine when booted.

System BIOS will more than likely have an option to set a password on startup as well as setup, ensure this is a good password and that it uses the maximum number of characters allowed, commonly eight. Set the BIOS to disable booting from the floppy disk drive, CD-ROM drive or other removable drive. If ever need to boot from a floppy disk user can always go back into the BIOS setup and enable it again.

Startup Files

To prevent somebody attacking computer system before Windows gets chance to start use these settings. This will prevent CTRL-BREAK during startup and disable the menu that allows booting to DOS.

1. Add the line SWITCHES=/F/N to CONFIG.SYS
2. Add the following lines to the top of AUTOEXEC.BAT

- CTTY NUL
- BREAK OFF
- ECHO OFF

Windows

The Windows-base operating systems often have a lot of flaws and security holes. In such, a Windows user should always download security patches from the Microsoft's home. This step is to minimize the Windows flaws. Windows Operating System can be tweaked using tweaking application in the market. Below are some tweaking guidelines.

1. Don't make the Boot-function key available; not allowing F4 key to boot from previous operating system; not showing boot menu; not displaying splash screen when booting
1. Do not logon automatically in the network-enable computer
2. Turn off password caching; do wipe any existing password caches named USERNAME.PWL. Force windows to require Alphanumeric passwords
3. Set a screensaver password and perhaps set a reasonably short time before screensaver kicks in, that way the machine is protected when users are away from desk
4. Always turn off the machine immediately after shutting down
5. Wipe the Windows NT/Windows XP swap file

2.2.2 Password Security Issue

Password protection has been used for several years to control access to mainframe computer systems. More recently, passwords have also been implemented in the Personal Computer and Local Area Network, LAN environments.

Computer password is personal key to a computer system. Passwords help to ensure that only authorized individual access computer systems. Passwords also help to determine accountability for all transactions and other changes made to system resources, including data. If users share password with a colleague or friend, they may be giving an unauthorized individual access to the system.

Authentication of individuals as valid users, via the input of a valid password is required to access any shared automated information system. Each user is accountable for the selection, confidentiality and changing of passwords required for authentication purposes. Since user is responsible for picking up his or her own password, it is important to be able to tell the difference between a good password and a bad one. Bad passwords jeopardize the information that they are supposed to protect. Good ones do not.

Passwords should be kept confidential and should never be shared. Passwords should not be written down. Never use the same password twice. In fact, good access control systems prevent from choosing a new password that is similar to the old one. When selecting a new password, choose the one that is quite different from previous password.

Passwords should be changed frequently. The shorter the life of a password, the better it is. Some systems force users to change their password at predetermined intervals. Passwords should be at least four characters in length. If the system allows a password longer than four characters, then it is recommended that to use a minimum of six characters. Longer passwords are harder for others to guess.

Passwords should contain a combination of alphabetic, numeric and special characters. Avoid using any dictionary words. Passwords should not be trivial, predictable or obvious. Obvious passwords include names of persons, pets, relatives, cities, streets and so on. Predictable passwords include days of the week, months, or a new password that has only one or two character different from the previous one. Trivial passwords include common words like 'secret', 'password', 'sex', 'computer', etc.

A good password is relatively easy to remember but hard for somebody else to guess. There are a variety of techniques that can be used to choose secure passwords. [11]

2.2.3 Computer Virus Security Issue

Computer viruses are malicious programs (or sets of instructions) which are designed to replicate themselves from one computer to another. Viruses may mutate over time if they are designed to change their own content as they replicate or if they are changed by someone who has access to the code. Viruses may also contain destructive code. These programs are generally designed to make changes to software that already exists on the system thus enabling them to replicate themselves, display messages, destroy data or software, or inhibit access by the user of the system. [12]

Computer viruses can be a serious threat to an organization since they usually result in loss of productivity. Viruses are a threat for a number of reasons, including:

1. Viruses can quickly spread to a large number of computer systems or networks.
2. Access to personal computers is usually not restricted. This means that anyone could introduce a virus, intentionally or otherwise, to the environment.
3. Viruses can mutate as they propagate through an environment. These types of viruses are very hard to detect and usually more harmful.
4. Although techniques for detecting and removing known viruses have been developed and are relatively cheap to acquire, the problem will not disappear since not everyone protects and monitors their system or takes the steps necessary to prevent infection.

Malicious Code

Malicious code is the general name for unanticipated or undesired effects in programs or program parts, caused by an agent intent on damage. It may be categorized to logic bomb, parasite, Trojan horse, virus and worms.

1. A logic bomb is a program that lies dormant until it is activated. It can be activated by anything that the computer system can detect.
2. A parasite is a piece of code that is added to an existing program and draws information from the original program.

3. A Trojan horse is a program that looks like a useful program that has an alternate agenda. [13]
4. A virus is a program that infects another program by replicating itself into the host program.
5. A worm is a program that is used as a transport mechanism for other programs.

The PC environment is not controlled like a mainframe computer environment. Sharing of Personal Computers, files and diskettes is quite common. There are a number of ways in which a computer can become infected, including:

1. Use of an infected diskette on a computer.
2. Software downloaded from the Internet and files attached to E-Mail messages may be infected with a virus.
3. Pirated computer games are frequently infected.

Do not attempt to remove the virus unless virus scanner software does it automatically. The best thing to do is to stop using computer and make sure that the infected computer and any media used on it are isolated until the problem is fixed. There are some very basic things to prevent virus infections. Safe computing is relatively easy to achieve. The following guidelines will help to practice safe computing.

Do not open e-mail attachments if are not expecting an attachment from someone know or trust. Many of the new generation viruses, such as "Melissa", come as attached files and are designed to spread themselves by opening MS Outlook address books and mailing themselves to all the addresses in the address book. Do not use illegal or "pirated" software on machine. Many times such software is infected. Do not use shareware unless absolutely sure that the software is free of viruses.

If download programs from Bulletin Boards or the Internet, or if receive files attached to E-Mail messages, scan them for viruses before use them. Many current releases of virus scanning software automatically scan for viruses when files are being downloaded over a network.

If bring any diskettes from home, scan them for viruses before using them on the computer at work. Do not install or play computer games on working computer. Games are commonly used as a way to spread computer viruses.

Back up files regularly (at least weekly). Make separate backups of data files and software and store backup diskettes or tapes in a safe and secure place away from computer. Backups may be the only way to recover any destroyed files. However keep in mind that backup files may also be infected.

When get new software for computer, make sure it is shrink-wrapped and check the diskettes before using them on the computer. Make sure that diskettes used to store software programs are write-protected. This prevents viruses from being copied onto such diskettes.

If "new" computer is in reality a recycled one that someone else used before, ask the individual installing computer to do a "low-level format" of the hard drive. This will destroy any viruses that may be on the hard drive as well as get rid of illegal copies of software. If "new" computer comes pre-loaded with software or if the hard drive is pre-formatted, scan the hard drive for viruses before starting to use the computer.

Do not boot computer with a diskette that has not been scanned for viruses even the diskette only contains data. Never booth computer from a diskette created on another computer. If booting computer from the hard drive, make sure that there isn't a diskette in the diskette drive. If computer is located in an open office area or if the computer is used by more than one individual, installs a security package that restricts use of the computer to authorized users.

Public-domain software should not be used until it is tested and labeled "clean". All software should be acquired from reputable dealers. Non-shrink-wrapped software may

be infected although there are known incidents involving spreading of viruses through shrink-wrapped software.

2.2.4 Computer Administration Issue

The system administrator has three main objectives:

1. Ensure that the system does its job effectively and efficiently
2. Ensure that the information stored on the system is secure from intentional or accidental destruction
3. Administer the system owner's rules for the use of the system.

To achieve these objectives the system administrator must understand more than just the structure and interaction of the hardware and software under their control. They must also understand the interconnected environment in which almost all current systems exist and the effects that environment has on function and performance of the local system. [14]

A contemporary computer system includes a number of hardware, software, and information elements that must work cooperatively if the system is to satisfy the needs of its users. [15] The main elements and their management functions are:

Fixed-disk drives

1. Control the grouping and subdivision of disk space.
2. Control the location of data and programs for optimum performance.
3. Control the amount of space allocated for different purposes.

Application programs

1. Control the use of sensitive or costly programs.
2. Install and performance-tune major applications.

Application data

1. Control access to sensitive data.
2. Ensure that appropriate backup measures are taken.

Individual computer processors and memory

1. Ensure that resources are used in accordance with the priorities of the organization.
2. Control access to the system by individuals and groups.
3. Tune the operating system for optimal use of the available resources.

Local area networks

1. Ensure that networks are tuned for optimum performance.
2. Control network addressing mechanisms.

Local terminals

1. Control the connection of terminals to processors.
2. Ensure that terminals and processors are set up for maximum performance.

Connections to other networks

1. Ensure that bridges and gateways to other networks are configured correctly.
2. Ensure that interaction with remote networks does not degrade local systems.

Access to and from remote systems

1. Control the access permissions in both directions.
2. Monitor and performance-tune the workload imposed by remote connections.
3. Control the methods and availability of access to remotely owned data.

2.3 Current Market Security Applications Study

There are numerous security related applications exist in the software market. The PESAT development has been undergone some of the security applications in the market. As the result, the reviewed of the security applications are presented here, with the commented description, strength and weakness of the respective application. PESAT

features on the basis are similar to those market security applications, with the modified features of those weaknesses which is defined and the new innovation of security measures.

2.3.1 PC Security by Tropical Software

Availability

A shareware innovated by Tropical Software that can be downloaded at http://www.tropsoft.com/.

Description

The PC Security for Windows offers comprehensive data security protection by performing the functionalities like File locking, Folder locking, Internet locking, Window locking., Shortcut and Program locking, System locking for the whole computer with different timers and Hot Key, Explorer Control, Restricted System, Intruder Detection with Alarm, Context Menu support in mouse right click to lock and unlock, Flexible and complete password protection, Drag and Drop support. [16]

Strength

The layout is arranged properly so that the user can locate the desired actions in ease. The application is a simple computer security tool that provides much great functionalities. It is best used by the novice user. Password to enter the security setting protects unauthorized people from access the computer.

Weakness

Some features have to wait the windows to reboot to active the security actions. This wastes user time. The user management module does not effectively manage the user. This means by the security measures is completely useless if the operating system adds in a new user. No part of auto antivirus definition update.

2.3.2 Microsoft Baseline Security Analyzer by Microsoft Corporation

Availability

A shareware innovated by Microsoft Corporation that can be downloaded at http://www.microsoft.com/.

Description

The Microsoft Baseline Security Analyzer checks computers running Microsoft Windows NT 4.0, Windows 2000, or Windows XP for common security configuration errors. It only provides the user with the administrator privileges for each computer to be scanned. Scans can be performed locally and remotely against Windows NT 4.0, Windows 2000, and Windows XP computers. [17]

Strength

Microsoft Baseline Security Analyzer checks many security settings during a full scan. It has numerous modules like Windows checks, IIS checks, SQL checks, and Desktop application checks. The Microsoft product is able to scan more than one computer using IP ranging. Thus, it is a convenient handy tool. The deep blue interface impresses the user with its simplicity. It functions only in the Windows administrative log-in mode.

Weakness

Only local scan can be performed in the Windows XP computer. It only list out the error, not to correct them. Scanning process consumes quite some time. Do not protect files and folder sensitive information. No part of auto antivirus definition update.

2.3.3 Advance Administrative Tools by G-Lock Software

Availability

A shareware innovated by G-Lock Software that can be downloaded at http://www.glocksoft.com/amlv/index.htm?source=AAToolsMoreProd.

Description

Advanced Administrative Tools is a multithreaded network and system explorer. It's a 12-in-1 utility, including Port Scanner, Proxy Analyzer, RBL Locator, CGI Analyzer, Email Verifier, Links Analyzer, Network Monitor, Process Monitor, Who-Is, System Info, Resource Viewer and Registry Cleaner. Its purpose is to accumulate data pertaining to network status and availability, using all of the latest development tools in network research. The application is security toolset that finds holes before intruders attack. The application should be a part of computer security toolkit. [18]

Strength

A complicated network monitoring tool provides most of the day-to-day network monitoring tasks. The interactive design would make it perfectly the absolute tool for the network administrator. It can scan port, verify email or perform registry cleaning. Very well design for the computer over networks. It has the Socks Proxy Analyzer database proxies with the same IP but different ports.

Weakness

It is not well suited for the personal computer usage. The usage of much computer jargon might confuse the user. It does not protect sensitive information and no authentication to use the application. This makes it not the securable application at the overall view point. No part of auto antivirus definition update.

2.3.4 Tweak XP by Total-Idea Software

Availability

A shareware innovated by Total-Idea Software that can be downloaded at http://www.totalidea.de.

Description

Tweak-XP was developed to combine both tweaking and optimizing features to increase the speed of Windows XP system. All settings can be done without having any former knowledge, since Tweak-XP informs the user in detail about each option. [19]

Strength

It optimizes Internet connection speed, tweaks hardware, optimizes the physical RAM, censors the execution of applications, protects and hides the content in the hard disk, creates virtual disk drives, control the windows start-up, checks the ZIP file integrity, synchronizes system time to an atomic clock, manage the internet history files, block the internet banners and pop-up, customize the start-menu and generate possible password. "System Restore Point" tends to recover from the undesired security measures apply in the system.

Weakness

The application can only apply to single user in multiple user platforms. Switching to another user would not apply the security measures. No authentication available when entering the application. No part of auto antivirus definition update.

2.3.5 Tweak Manager by Win-Guide Network

Availability

A shareware innovated by Win-Guide Network that can be downloaded at http://www.karchitects.com/.

Description

Win-Guides Tweak Manager uses a Windows-XP interface to provide user access to over 800 Tweaks for the Microsoft Windows operating systems. These tweaks can be applied to such areas as Windows core operating system features, Hardware, Network, Security and Software. [20]

Strength

A powerful security tool that covers numerous security tweak in many areas such as desktop, networking, computer login and authentication, hardware, software and file system. Beautify with its authentic interface and the simplicity non-jargon languages, the application tends to draw the attention of the computer user in the market.

Weakness

No login to the application makes it an insecure application to the administration. Anyone can change the settings easily, not restricted by the tweak completely. No part of auto antivirus definition update.

2.3.6 Hyena NT Administration by Giant Technology

Availability

A shareware innovated by Giant Technology that can be downloaded at http://www.giant-technologies.com/hyena/.

Description

Hyena provides a single interface for managing all aspects of NT/2000/XP based networks. As well as containing the standard functionality of Server Manager, User Manager and other, Hyena adds extra abilities of its own. Hyena makes the systems management fast and simple. It contains functionality for managing MS Exchange and Terminal Server users from the same central location. [21]

Strength

It supports for Active Directory Administration. It is best for edit the registry management. It contains a powerful user management control system. Hyena provides access to every aspect of server administration. Hyena also incorporates its own more powerful event viewing mechanism. It can be used across multiple operating system platforms with vivid interface design.

Weakness

The interface design is too professional-like which might not very well suit for typical user as there exists numerous computer jargons. More to network security measures than computer security measures. No part of auto antivirus definition update. No authentication to enter the application makes it very easy to change the pre-configure security setting.

2.3.7 Security Administrator Tool by Ixis Research

Availability

A shareware innovated by Ixis Research Centre that can be downloaded at http://www.softheap.com/secagent.html.

Description

Security Administrator for Windows is to restrict access to the computer. The password-protected security utility can impose a variety of access restrictions to protect privacy and stop others from tampering with user desktop. [22]

Strength

The application denies access to each individual component of several Control Panel applets, including Display, Network, Passwords, Printers, and System. It disables boot keys, DOS windows, Registry editing, and network access. It hides desktop icons, individual drives, Start menu items, or even the entire taskbar. User can also apply password protection to Windows and restrict users to running specific applications only. Security restrictions can be applied universally or just for specific users. User will find the program interface very easy to negotiate. Excellent online help is available.

Weakness

Some features have to wait the windows to reboot to active the security actions. This wastes user time. The interface design is too raw and simple. No part of auto antivirus definition update. No authentication to enter the application makes it very easy to change the pre-configure security setting.

2.4 Strengths and Weaknesses

From reviewing some of the current security and administration application in the market, remarkable strengths and noticeable weaknesses has been drawn out as for the PESAT application development references. PESAT application is adopting the strengths of the current application in the development; on the same time, more

applicable measurements have been adhered in the PESAT tool to overpower the weaknesses of the current application. Below is the listing of current security administration tool strengths and weaknesses in commonality.

Strengths in Common:

1. It provides numerous yet sufficient administration and security settings for the user.
2. Most of the administration tweakings and security measurements are the hidden functionalities in the Windows XP operating system.
3. It tends to apply more administration tweakings as well as security measurements by manipulating the registry values.
4. The interface design is interactive, impressive, inspirational and notable with the well-structured sufficient documentation on site.

Weaknesses in Common:

1. Most applications differentiate the security element with the computer administration.
2. It overlooks the vital part of virus scanner element in the security administration.
3. It does not provide authentication to the application which allow any user to alter the computer setting easily.
4. It supports only one user profile in the multiple-support operating system. The security measurements would not applicable in another user profile in multiple-support operating system.
5. It only can be deployed in one operating system. Rarely to see the application that is able to span across more than one operating system.
6. The installed folder and files within operating system is not encrypted and can be viewed independently, edit or even delete the files completely.
7. It is visible and can be uninstalled using the add remove program in the control panel.

8. It tends to use much computer jargons that might confuse the beginner computer user.
9. Most of the applications are distributed as commercialize shareware, where the user has to pay for the desire application.

2.5 Similarities and Differences

PESAT application is developed with remarkable similarities as well as noticeable differences with the current application on security and administration. The below is the listing of PESAT application similarities and differences compare to the market application.

Similarities:

1. PESAT provides security and administration measurements in general mostly the same with the market tools, like desktop management, internet management, file and folder security, add remove application, and short cut to turn down the computer.
2. It is used for only one user profile, thus not very applicable in the networking environment.
3. A simple user-friendly yet artistic interface with the excellent documentation guiding user confidently to append or amend security and administration measurements in their computer system.
4. Only to be used in one operating system, namely Windows XPProfessional environment.

Differences:

1. Virus scanner security is the unique measurements in the PESAT application where it intelligently updates the virus definition to the virus scanner utility.
2. It is not listed in the control panel and no existence of uninstaller within the application. User can only uninstall PESAT application inside the application. The installed folders and files can not be viewed within the explorer.
3. It contains authentication with the password to enter the application.
4. Minimum usage of computer jargons and hard-technical computer terms.
5. It purposely distributed as a freeware to raise the user attention of the computer security and administration issues.

2.6 Application Development Tool Study

PESAT application is developed using current market programming technologies and specially suit on certain operating system. Consequent part is the introduction to some operating system platforms and numerous programming technologies.

2.6.1 Application Operating System Platform

The operating system that is reviewed is

1. Windows NT Server Operating System
2. Windows XP Professional Operating System
3. UNIX Operating System
4. LINUX Operating System

Windows NT Server Operating System

Microsoft Windows NT is one of the powerful operating system for business computing. It combines the ease-of-use of Windows 95 with the power and reliability of Windows NT. NT is also a powerful operating system that reliable, secure, multithreaded, symmetric processing, and support client/server system.

There is an extensive security support in NT. NT can control the access control of user in accessing certain file or application. This can use for implemental the access control for the Attendance Management System. Besides, NT supports a wide range of networks protocol and Remote Access Protocol. These make it easy for us to develop the distributed applications. Windows NT Server is a complete platform available for building and hosting web based applications. It is the best platform to publish and share information securely over corporate Intranet and Internet. It is so reliable that when an application has problem it doesn't crash the whole program.

Windows NT allows Object Linking and Embedding (OLE). It can combine the information from several applications into one compound document using the special

OLE capabilities of window-based application. Windows NT also enables the capabilities of integrating applications on a single computer or even accesses multiple computers.

Windows XP Professional Operating System

Built on NT Technology and an easy-to-use, familiar Windows 98 user-interface, Windows XP Professional makes users more productive. Its integrated Web capabilities and broad support for mobile computers and hardware devices makes Windows XP the easy way for users to connect to the Internet anywhere and anytime.

No matter where user is working, the computer will be easier to use and to manage, because Microsoft Windows XP Professional is more compatible and more powerful than any workstation. Statistic from Microsoft Official Web-site reveals that Windows XP users have been dominated 45% out of Windows operating system family.

Easier to use

With Windows XP Professional, user has faster access to information, and able to accomplish tasks more quickly and easily. Windows XP Professional makes it easier to:

1. Work with files.
2. Find information.
3. Personalize computing environment.
4. Work on the Web.
5. Work remotely.

Easier to manage

User and network administrators can work more efficiently now, because many of the most common computer-management tasks are automated and streamlined with Windows XP Professional. With Windows XP, the workstation will be easier to:

1. Set up
2. Administer

3. Support

More compatible

Windows XP Professional offers increased compatibility with different types of networks and a wide array of legacy hardware and software. Windows XP provides:

1. Improved driver support.
2. Increased support for new-generation hardware and multimedia technologies.
3. Integration of the new Euro currency symbol.

More powerful

For all users' computing needs, Windows XP Professional provides:

1. Industrial-strength reliability.
2. The highest level of security.
3. Powerful performance.

Comparison between Windows XP Professional and Windows NT Server

Client Performance

In addition to the many new desktop features that have been added to Windows XP Professional such as Plug and Play, power management, single worldwide binary for all languages, IntelliMirror management technologies, and many others, the performance on the desktop is significantly faster than Windows 95 & 98 and comparable to Windows NT Workstation 4.0. By optimizing fundamental system services such as memory management, registry access, and disk I/O, Windows XP is able to deliver the added benefit of new features while delivering the performance that users expect.

Networking Performance

Windows XP is Gigabit ready. This means that Windows XP is capable of sustaining the highest levels of network throughput using Gigabit adapters. For example, using network adapters designed to utilize the advanced TCP/IP features in Windows XP such as TCP/IP checksum offloading and large send support, Windows XP can deliver up to

25% better network throughput than Windows NT Server 4.0 and sustain close to 4 gigabit per second (Gbps) of data throughput.

File & Print Server Performance

By improving many aspects of the file serve components such as increasing the virtual size of the file cache form 496MB to 960MB, optimizing the SMB redirector on Windows XP, and optimizing the Windows NT File System (NTFS), Windows XP provides users with a better performing and scalable file server solution. Using the Net-Bench benchmark, Windows XP Server 2003 with Windows XP Professional-based clients provides up to 20% better performance than Windows NT Server 4.0 with Windows NT Workstation 4.0-based clients. In some cases, especially where the file shares are stored on a single partition, Windows XP provides up to 2-times better file server performance than Windows NT Server 4.0.

As a print server, Windows XP is capable of processing up to 3-times more pages per second when compared to Windows NT Server 4.0. Moreover, as the chart to the left shows, print performance doesn't degrade on Windows XP Server 2003 as additional printers are added. Furthermore, Windows XP can perform the rendering in 1/3 of the time it took on Windows NT Server 4.0 and requires 1/3 of the CPU resources.

UNIX Operating System

The first version of UNIX was created in 1969 by Kenneth Thompson and Dennis Ritchie, system engineers at AT&T's Bell Labs. It went through many revisions and gained in popularity until 1977, when it was first made commercially available by Interactive Systems Corporation.

UNIX is an increasingly popular operating system found on multi user installations. It is currently available on personal computers instead of used on minicomputers and workstations in the academic community traditionally. UNIX, like other operating systems, is a layer between the hardware and the applications that runs on the computer.

It has functions that manage the hardware and functions that manage the execution of applications. UNIX can be used for:

1. Sending and receiving e-mail, forwarding mail, redirecting mail, mapping a particular mail group to a list of specific users.
2. Storing files, including users' personal files as well as publicly accessible software archives.
3. Managing centralized databases and serving information to users remotely.
4. Running a web server and storing web pages. The UNIX machines are normally left on 24 hours a day.
5. Implementing shared network file systems.
6. Teaching. UNIX provides an environment for the controlled management of modules and courses.
7. The Common Gateway Interface (CGI) – CGI scripts can be set up so that programs on machines can be executed across the World Wide Web. This allows information services to be provided in real time.
8. Remote services. Computers running UNIX normally support certain remote services, allowing users to request information from the computer without actually logging in.
9. Software development. All UNIX systems have a C compiler, and many other languages such as Prolog, Common Lisp and FORTRAN are available as well.
10. Computational Mathematics. Under UNIX, it is very easy to set up computational jobs that will be run overnight or for many days. Output can be written to a file and viewed at a later time.

The benefits and weaknesses of UNIX can be concluded as the following:

Benefits

1. UNIX is consistent in treating files. It is very easy for the users to work with files because users do not need to learn special command for every new task.
2. UNIX is not known only for its longevity and versatility as an operating system, but also for the variety and number of utility programs that called tool.

3. It is a powerful and mature operating system and network-based application.

Weaknesses

1. UNIX needs very powerful workstations. Therefore it is not cost effective to use.
2. UNIX is very expensive.

Linux Operating System

Linux is an operating system that was initially created as a hobby by a young student, Linus Torvalds, at the University of Helsinki in Finland. Linus had an interest in Minix, a small UNIX system, and decided to develop a system that exceeded the Minix standards. He began his work in 1991 when he released version 0.02 and worked steadily until 1994 when version 1.0 of the Linux Kernel was released. The current full-featured version is 2.4 (released January 2001) and development continues. Linux is another version of UNIX-based operating system that is written from scratch to avoid license fees entirely. The following are the important features of Linux:

1. Linux is a real multitasking system that allows multiple users to run programs on the same system at once. Linux is also a full 32-bit operating system that utilizes the special protected-mode features of Intel 80386 and later processes and their work-alike.
2. The X Window system is a very powerful graphics interface, supporting many applications.
3. Linux is built in networking support. It uses standard TCP/IP protocols, including Network File System (NFS) and Network Information Service (NIS – formerly known as YP).
4. Linux implement shared libraries that allowing programs use standard subroutines to find the code for those subroutines in the libraries at runtime.
5. Linux supports a wide range of free software written by the GNU Project, including utilities such as GNU C and C++ compiler, gawk, gruff and many more. Many of the essential system utilities used by Linux are GNU software.

6. Linux is compatible with the IEEE POSIX.1 standard. It supports many important features of other UNIX standards.
7. Linux is fault-tolerant. With Apache as the primary application for those servers, they have proven to be practically immune to the recent explosion of viruses that have plagued e-mail and the Internet.
8. The Linux kernel uses no code from AT&T, nor any other proprietary source, other organizations, such as commercial companies, the GNU project, hackers and programmers from all over the world have developed software for Linux.
9. Because of the available source code and the ability for users to modify, Linux is not as secure as other system if an ever-expanding group of hackers who want to get their hands dirty with others' Linux-based system.
10. Lower cost than most over Windows NT system and UNIX clones systems, as Linux is freely available on the Internet.

The benefits and weaknesses of Linux can be concluded as:

Benefits

1. Linux is as stable as UNIX.
2. Highly cost-effective ability to scale the size of the site as traffic grows.
3. It is developed under the GNU General Public License and its source code is freely available to everyone.

Weaknesses

1. It is developed worldwide, therefore lack of proper organized support.
2. Linux is inherently unsafe because every malicious cracker in the universe has the source code to the site.
3. Linux is missing many pieces required to build a real application. Those pieces are problematic.

2.6.2 Application Programming Languages

PESAT application is surely built by numerous blocks of programming codex. It might manipulate several types of programming languages. Consequently, several programming languages have been reviewed.

1. Visual Basic Programming Language
2. Visual Basic NET Programming Language
3. JAVA Programming Language
4. Windows Management Instrumentation, WMI Script
5. Windows Script Host, WSH Script
6. Active Directory Service Interface, ADCL Script

Visual Basic 6 Programming Language

Microsoft Visual Basic is the fastest and easiest way to create applications for Microsoft Windows. Whether users are an experienced professional or brand new to Windows programming, Visual Basic provides them with a complete set of tools to simplify rapid application development. [27]

The "Visual" part refers to the method used to create the graphical user interface (GUI). Rather than writing numerous lines of code to describe the appearance and location of interface elements, user can simply add pre-built objects into place on screen. If users have ever used a drawing program such as Paint, they already have most of the skills necessary to create an effective user interface.

The "Basic" part refers to the BASIC (Beginners All-Purpose Symbolic Instruction Code) language, a language used by more programmers than any other language in the history of computing. Visual Basic has evolved from the original BASIC language and now contains several hundred statements, functions, and keywords, many of which relate directly to the Windows GUI. Beginners can create useful applications by learning just a few of the keywords, yet the power of the language allows professionals to accomplish anything that can be accomplished using any other Windows programming language.

The Visual Basic programming language is not unique to Visual Basic. The Visual Basic programming system, Applications Edition included in Microsoft Excel, Microsoft Access, and many other Windows applications uses the same language. The Visual Basic Scripting Edition (VBScript) is a widely used scripting language and a subset of the Visual Basic language.

Whether the goal is to create a small utility for personal or work group, a large enterprise-wide system, or even distributed applications spanning the globe via the Internet, Visual Basic has the tools for every need.

1. Data access features allow creating databases, front-end applications, and scalable server-side components for most popular database formats, including Microsoft SQL Server and other enterprise-level databases.
2. Active-X technologies allow using the functionality provided by other applications, such as Microsoft Word word-processor, Microsoft Excel spreadsheet, and other Windows applications. User can even automate applications and objects created using the Visual Basic programming technology.
3. Internet capabilities make it easy to provide access to documents and applications across the Internet or intranet from within application, or to create Internet server applications.
4. The finished application is a true .exe file that uses a Visual Basic Virtual Machine that can freely distribute.

Some of other good features of Visual Basic programming Language:

1. User-defined types in Public Methods can be arguments or return types of public properties and methods.
2. Functions and properties procedures will be able to return arrays.
3. Variable-sized arrays can now appear on the left side of an assignment. Fixed-sized arrays cannot appear on the left side of an assignment, although they can appear on the right.

4. File System Objects offers a streamlined set of routines to traverse the file system and create text files and directories.
5. The Dictionary object overcomes some of the Collection object's limitations and is easier to use because it generates fewer run-time errors for common operations and is more flexible.
6. This new data access technology features a simpler object model, better integration with other Microsoft and non-Microsoft technologies, a common interface for both local and remote data access, remote-able and disconnected recordsets, a user-accessible data binding interface, and hierarchical recordsets.
7. The Data Environment designer provides an interactive, design-time environment for creating ADO objects. These can be used as a data source for data-aware objects on a form or report, or accessed programmatically as methods and properties exposed off of the Data Environment object. The Data Environment designer supports all the functionality of Visual Basic's User-Connection designer, as well as additional features, including drag and drop, hierarchies, grouping, and aggregates.
8. A new OLEDB-aware data source control that functions much like the intrinsic Data and Remote Data controls, in that it allows user to create a database application with minimum code.
9. Visually create and modify database schemas and queries: Create SQL Server and Oracle database tables drag and drop to create views, and automatically change column data types.
10. SQL Editor allows user to add new stored procedures to existing SQL Server and Oracle databases. User can write triggers using the editor too.
11. Data Object Wizard automates creating middle-tier objects bound to the Data Environment or User Controls.
12. File System Objects offer a streamlined set of routines to traverse the file system and create text files and directories.
13. Format Objects provide two-way conversion of data between a database and a bound control: When a value is read from a database, the Format objects add formatting appropriate to the bound control.

Visual Basic NET Programming Language

Visual Basic .NET has many new and improved language features, such as inheritance, interfaces, and overloading that makes it a powerful object-oriented programming language. As a Visual Basic developer, user can now create multithreaded, scalable applications using explicit multithreading. Other new language features in Visual Basic .NET include structured exception handling, custom attributes, and common language specification (CLS) compliance. [28]

The CLS is a set of rules that standardizes such things as data types and how objects are exposed and interoperate. Visual Basic .NET adds several features that take advantage of the CLS. Any CLS-compliant language can use the classes, objects, and components created in Visual Basic .NET. And as a Visual Basic user, they can access classes, components, and objects from other CLS-compliant programming languages without worrying about language-specific differences such as data types. CLS features used by Visual Basic .NET programs include assemblies, namespaces, and attributes.

Visual Basic .NET supports many new or improved object-oriented language features such as inheritance, overloading, the 'Overrides' keyword, interfaces, shared members, and constructors.

Also included are structured exception handling, delegates, and several new data types.

Inheritance

Visual Basic .NET supports *inheritance* by allowing user to define classes that serve as the basis for derived classes. Derived classes inherit and can extend the properties and methods of the base class. They can also override inherited methods with new implementations. All classes created with Visual Basic .NET are inheritable by default. Because the forms are really classes, user can use inheritance to define new forms based on existing ones.

Exception Handling

Visual Basic .NET supports *structured exception handling*, using an enhanced version of the *Try...Catch...Finally* syntax supported by other languages such as C++. Structured exception handling combines a modern control structure (similar to *Select Case* or *While*) with exceptions, protected blocks of code, and filters. Structured exception handling makes it easy to create and maintain programs with robust, comprehensive error handlers.

Overloading

Overloading is the ability to define properties, methods, or procedures that have the same name but use different data types. Overloaded procedures allow user to provide as many implementations as necessary to handle different kinds of data, while giving the appearance of a single, versatile procedure.

Overriding Properties and Methods

The *Overrides* keyword allows derived objects to override characteristics inherited from parent objects. Overridden members have the same arguments as the members inherited from the base class, but different implementations. A member's new implementation can call the original implementation in the parent class by preceding the member name.

Constructors and Destructors

Constructors are procedures that control initialization of new instances of a class. Conversely, *destructors* are methods that free system resources when a class leaves scope or is set to *Nothing*. Visual Basic .NET supports constructors and destructors using the *Sub New* and *Sub Finalize* procedures.

Data Types

Visual Basic .NET introduces three new data types. The *Char* data type is an unsigned 16-bit quantity used to store Unicode characters. It is equivalent to the .NET Framework *System. Char* data type. The *Short* data type, a signed 16-bit integer, was named *Integer* in earlier versions of Visual Basic. The *Decimal* data type is a 96-bit signed integer

scaled by a variable power of 10. In earlier versions of Visual Basic, it was available only within a *Variant*.

Interfaces

Interfaces describe the properties and methods of classes, but unlike classes, do not provide implementations. The *Interface* statement allows user to declare interfaces, while the *Implements* statement lets user writes code that puts the items described in the interface into practice.

Delegates

Delegates — objects that can call the methods of objects on user behalf — are sometimes described as type-safe, object-oriented function pointers. User can use delegates to let procedures specify an event handler method that runs when an event occurs. User can also use delegates with multithreaded applications.

Multithreading

Visual Basic .NET allows user to write applications that can perform multiple tasks independently. A task that has the potential of holding up other tasks can execute on a separate thread, a process known as *multithreading*. By causing complicated tasks to run on threads that are separate from user interface, multithreading makes applications more responsive to user input.

JAVA Programming Language

A Java program is made up of pieces called classes. Each piece of classes can be program by ourselves to form a Java Program. But most Java programmers take advantage of rich collections of existing classes in Java class libraries. Class libraries are provided primarily by compiler vendors, but there are many class libraries that are supplied by independent software vendors. Java systems generally consist of several parts: an environment, the language, the Java Applications Programming Interface (API), and various class libraries. [29]

Java is an ideal companion to XML: both languages share a similar historical background (C++, SGML). Both Java and XML have goals of simplicity, portability, and flexibility and both continue to be developed in groups that involve industry, development community and academia (W3C, JCP). The Java Software Platform has a comprehensive collection of core APIs specifically targeted at building XML-based applications.

Java was invented by Sun Microsystems developer James Gosling as a better way to create computer programs. Gosling was unhappy with the way that the C++ programming language was a project he was doing, working on so he created a new language that did the job better. It's a matter of contentions debate whether Java is superior to other programming languages, of course, but the amount of attention paid to the language today shows that it has a large number of adherents. Java is the best language; it definitely is a great language to learn today. The language has become a major part of the Internet's future.

One of the biggest advantages of learning Java is that a developer can use it on the WWW. It can be used to create animated graphics, present text in new ways, play games, and help in other interactive efforts.

Another important advantage is that Java requires an organized approach in order for programs to work. The language is very particular about the way that programs must be written, and it stops if programmers do not follow all of its rules. When a developer starts to write Java programs, he must not see the language choosy behavior as an advantage. He will write a program and will have several errors to fix before the program will be finished. Some of his fixes might not be correct, and they will have to be redone. If he does not structure a program correctly as he is writing it, other errors will result. The positive side of this extra effort is that the programs will be more reliable, useful and error-free.

Windows Management Instrumentation, WMI

Windows Management Instrumentation, WMI is a unified architecture for describing, accessing, and instrumenting objects. Part of this architecture is a large repository of CIM classes that WMI can use to carry out management tasks on specific objects. WMI class is template for a type of managed object. WMI classes are based on the Common Information Model (CIM), as implemented in Windows Management Instrumentation (WMI). It is also a set of Active Directory Service Interface (ADSL) and Windows Script Host (WSH). Below is the listing of some general WMI description:

1. Scripts can access all WMI classes for hardware and software objects.
2. WSH scripts can perform operations on file system objects, manipulate network printers, or change environment variables.
3. ADSI scripts provide access to Active Directory objects.
4. Both WSH and ADSI scripts access objects and allow procedures not available through batch files.
5. Any computer information can be manipulated through WMI using scripts.
6. Scripts can be written in any scripting language that supports Microsoft ActiveX script hosting, including Visual Basic Scripting Edition (VBScript), Microsoft JScript, and Perl.
7. WSH, Active Server Pages, and Internet Explorer can all host WMI scripts.

WMI makes Windows XP extremely manageable using a single consistent, standards-based, extensible and object-oriented interface. Also, any application or script accessing WMI data can do so on the local machine or remotely in a seamless way. And, it's not only for Windows XP. WMI is available for Windows 95, Windows 98, and Windows NT 4.0. There are several key features in WMI that will be valuable in solving the complex management tasks IT administrators are challenged with today:

Uniform Scripting API

All managed objects are defined under a common object framework based on the CIM object model. Scripts only need to use a single API, WMI, to access information for numerous disparate sources.

Remote Administration

Objects managed within WMI are by definition available to applications and scripts both locally and remotely. No additional work is needed to manage remote objects.

Discoverability and Navigation

Applications and scripts are able to discover what information is available about a system by enumerating the classes that are available. Relationships between related objects can be detected and traversed to see how one managed entity affects another.

Query Capability

WMI treats its managed data much like a relational database and allows for SQL queries to be submitted in order to filter and focus requests for data to only that of interest.

Powerful Event Publication and Subscription

Events can be requested for virtually any change in the managed objects in the system, regardless of whether they support an internal event capability. Event subscribers are able to request notification of very specific events based on their particular interests rather than only being able to get events that were predefined by the original developers. Also, a very flexible architecture allows virtually any user-defined action to be taken upon the receipt of a given event.

Windows Script Host, WSH

Windows Script Host is built into Microsoft Windows 98, 2000, and Millennium Editions. By default, two script engines are included with WSH: Visual Basic scripting and JavaScript scripting. WSH can be called in two ways:

1. Windows based (WScript.exe);
2. Text based (CScript.exe);

The WSH execution environment includes wscript.exe, a windows-based host; cscript.exe, a console-based host; and wshom.ocx, the WSH Object Model runtime library. WSH also includes scripting engines: Microsoft JScript 3.0 (jscript.dll) Microsoft VBScript 3.0 (vbscript.dll). PearlScript, Python (not included by default) And makes use of Object Linking and Embedding (OLE) automation servers, like:

1. ADSI;
2. Excel;
3. Word;
4. Active Socket (ActiveXperts Software product);

WSH can handle different source files, and uses the Registry to map source scripts to the appropriate scripting engine. For instance, a source file with the .JS extension will be mapped to the JavaScript Engine. [25]

Wscript.exe and cscript.exe provide the interface between a script source and a registered script engine (i.e., interpreter). The wscript.exe and cscript.exe hosts are language independent. User can write scripts in any language that is registered with WSH. When user double-click a filename with a '.js', '.vbs', or other registered extension, wscript.exe (the default execution host) executes the script by passing it to one of the registered script engines. WSH selects the appropriate script engine based on the script extension. This information is written to the HKEY_CLASSES_ROOT Registry hive during the script engine registration process. User can change the default execution host to cscript.exe, or can specify the desired host via the Run dialog box or the command line at the time invoking the script. For example, from the command line user can enter

1. cscript c:\myscript.js
2. wscript c:\myscript.vbs

Wscript.exe and cscript.exe also provide several runtime options, including an interactive or batch-mode switch and a maximum number of seconds that a script can run. Both hosts support interactive and batch modes. But cscript.exe's interactive mode uses a command-line interface and wscript.exe's interactive mode uses simple graphical dialog boxes; using either host in batch mode is easy.

Active Directory Service Interface, ADSL

Active Directory Service Interface (ADSI) enables system administrators and developers of scripts or applications to easily query for and manipulate directory service objects. ADSI present a single set of directory service interfaces for managing network resources from different directory services. Administrators and developers can use ADSI to manage the resources in a directory service, regardless of which network environment contains the resource.

Scripts written to ADSI will work with any directory service that offers an ADSI provider. For example, with ADSI, applications can access LDAP, NDS, the Active Directory service, and other directories with ADSI interface as long as the appropriate service providers are available.

The standard ADSI providers are found within multiple namespaces - typically directory services for various network operating systems. Providers enable communication between the server and client. ADSI includes providers for:

1. Windows NT
2. Lightweight Directory Access Protocol (LDAP)
3. Windows XP Active Directory (AD)
4. Novell NetWare Directory Services (NDS)
5. NetWare 3 bindery (NWCOMPAT)

2.7 Summary

The literature review provides a resourceful of information about the security and administration issue to the developer. This is carried out as the research before the further development activities precede. The gathered information will be the consideration during the PESAT development so that it would not bias from its scope. The current application in the market similar to the PESAT application is shown here, with respective commented strengths and weaknesses compare to the PESAT application. The application development tools is also been reviewed. Following the next chapter is the methodology of software process model.

CHAPTER 3
METHODOLOGY

3.1 Introduction to Methodology

The system development methodology is a sequence of operations and procedures that are used to built up a system. This methodology is a very formal and precise system development process that defines a set of activities, methods, best practices, deliverables, and automated tools for system developers to develop and maintain most or all information systems and software.

Typically, the system development life cycle can be used to represent and define the methodology of the system development. Figure 3.1 denotes a typical system development process model for a general system development.

This part of the deliverable is concerned with a review of the methodology to be used in system design part of the PESAT project. Methodology ensures that a consistent reproducible approach is applied to PESAT development. It reduces the risk associated with shortcuts and mistakes. And it also produces complete and consistent documentation.

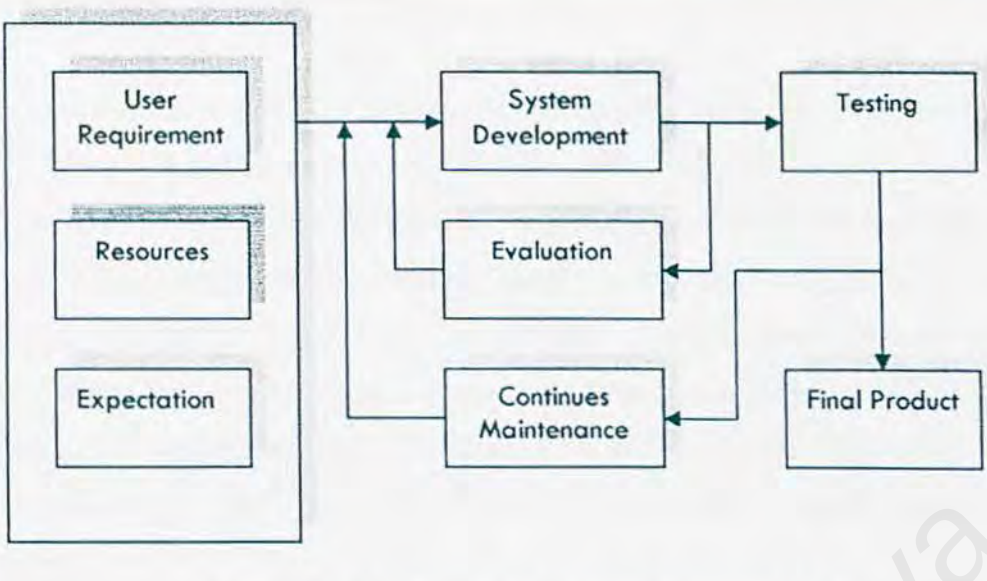


Figure 3.1: System Development Process Model

3.2.1 Concept of Methodology

Scientifically speaking a methodology is a "branch of philosophy" that deals with the science of method or procedure; or to put it differently a system of methods and rules applied in science. This reveals that a method is part of a methodology: a way of thinking, but more than that a way of doing. It provides the tool how to tackle problems using what techniques.

A method therefore enables one to follow a certain line, where its basis is laid down in the way a problem is encountered. Following a line rather than being forced to follow that line, as system development is a creative process that ultimately cannot be forced into rules and procedures without affecting the quality of the system. The reason is quite simple: there is no method that perfectly suits the needs of a system development project.

In general a method must fulfill two basic requirements.

1. Effective support of design process
2. Effective support of design process

Effective support of design process

The method must provide the means to identify the different steps in the development. This is many times known as the why-what-how chain: why do we need what and how will we achieve it" or in other words from system logic reasoning, semantic modeling towards syntax specification. The method should provide the means to set boundaries to system environment. It is required to set boundaries and only to take relevant aspects into consideration. An immediate consequence of the previous requirement is that the method should support stepwise refinement in the design process. The concept of decomposition of a total system into subsystems is required in order to reduce complexity.

Efficient control of project

The method should provide tools to an efficient control of a project. Most of the times a method uses the well-known concepts of activity planning, deliverable and milestone definition in the different stages of a system development project, no matter it applies to the design phase or the implementation phase.

3.2.2 Software Development of Process Model

There are several types of software development process model that are commonly deployed in the software development process, namely,

1. Waterfall Model
2. Waterfall Model with Prototyping
3. V Model
4. Prototyping Model
5. Rapid Application Development
6. Operational Specification Model
7. Transformation Model
8. Phased Development using Increments and Iterations
9. Spiral Model

3.3 Rapid Application Development

Overview

RAD is a methodology for compressing the analysis, design, build, and test phases into a series of short, iterative development cycles. This has a number of distinct advantages over the traditional sequential development model. RAD projects are typically staffed with small integrated teams comprised of developers, end users, and IT technical resources. Small teams, combined with short, iterative development cycles optimize speed, unity of vision and purpose, effective informal communication and simple project management. [25][26]

In the development of Personal Security Administrative Tool (PESAT), an iterative rapid application development (RAD) method is deployed. This methodology is based on solid requirements and design at first, followed by user feedback and rigorous testing to assure that the software development is on the successive path.

To catch any potential problem early in the development life cycle, both the processes and architecture are structured to allow for independent testing, apart from ongoing development and integration efforts. Testing is done on each functional unit, then on the unit as it is integrated into the complete system. This is done again after all units are integrated and after deployment. This architecture ties directly into supporting the iterative rapid application development (RAD) process.

Overall Conceptual

RAD methodology emphasizes extensive user involvement in the rapid and evolutionary construction of working prototypes of a system to accelerate the system development process. RAD is sometimes called a spiral approach because of the repeatedly spiral through the phases to develop a system in various degrees of completeness and complexity.

Quality is a primary concept in the RAD environment. Systems developed using the RAD development path meet the needs of their users effectively and have low maintenance costs.

The quality of a system is defined as the degree to which the system meets business requirements (or user requirements) at the time it begins operation. This is fundamentally different from the more usual definition of quality as the degree to which a system conforms to written specifications.

Rapid development, high quality and lower costs go hand-in-hand if an appropriate development methodology is used.

The basic ideas of RAD are:

1. To accelerate the requirements analysis and design phases through an iterative construction approach.
2. To reduce the amount of time until the users begin to see a working system.

RAD uses prototype to accelerate requirements analysis and system design.

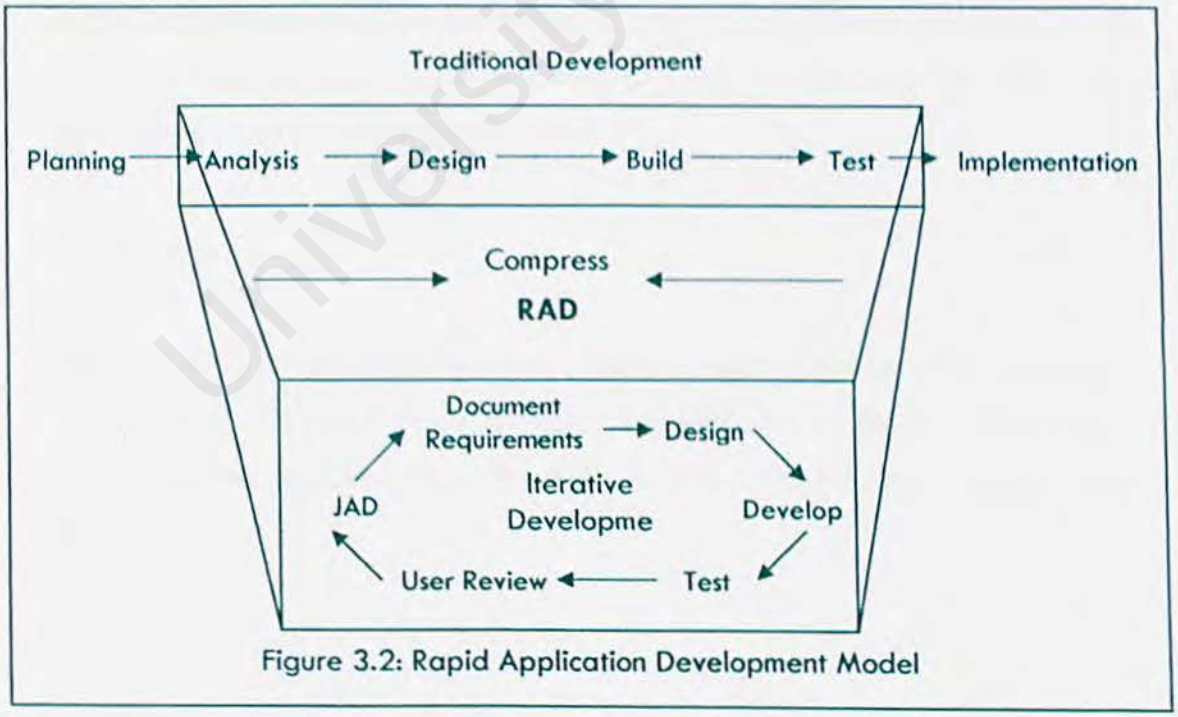


Figure 3.2: Rapid Application Development Model

Joint Application Design, JAD

Joint Application Design, or JAD, is a process originally developed for designing a computer-based system. It brings together business area people (users) and IT (Information Technology) professionals in a highly focused workshop. The advantages of JAD include a dramatic shortening of the time it takes to complete a project. It also improves the quality of the final product by focusing on the up-front portion of the development lifecycle, thus reducing the likelihood of errors that are expensive to correct later on

The fundamental principle of JAD development is to ensure the cooperatively work on defining the requirements for the project, setting priorities for development, examining and balancing options and quickly making decisions that meet the user's needs and are technically optimal. This minimizes the time necessary to develop the detailed requirements for any project, and more importantly, improves understanding, and tends to reduce rework, confusion, and other downstream problems that traditionally cause development time overruns.

PESAT application development integrates JAD process by having discussion with experienced lecturers and software developer. They provides many excellent ideas especially in the PESAT modules built-up.

Prototyping

Then the RAD process is implemented, where a rough prototype of the product is created to validate that it works as expected and delivers the desired functionality. When the prototype is approved, then the development is refined to create a robust, fully functional system.

Iterative Process

Iteration allows for effectiveness and self-correction. Studies have shown that human beings almost never perform a complex task correctly the first time. However, people are extremely good at making an adequate beginning and then making many small refinements and improvements. Iterative process reduces the amount of time until the users begin to see a working system.

3.3.1 Prototyping

A prototype is a smaller-scale, representative or working model of the users' requirements or a proposed design for an information system.

It is a process of iterative design and implementation of (a part of) the system with the objective to identify quickly insecurities. It can deal with unstable specifications because the primary goal is not to implement a full blown operational system but to identify and clarify problem areas, which were insecure when the project started. It can reveal a (partly) working system in a fairly fast time, so that feedback can be given at the soonest point in time.

The chances to involve final users - a necessary step in the development to obtain some guarantee for final acceptance of the system - increase because users will obtain already early a "look and feel" opportunity. Experience of users can be used and dealt with within the course of system development and implementation.

It reveals also the opportunity to manage the complexity of the system. Applying stepwise refinement reduces complexity and improves quick feedback because even early versions of tiny elements of the system can be released for that goal once the framework is implemented (first prototype of the system). And it does not hinder the implementation of the concept of modularity.

It will be the vehicle for developing the full requirements for the system, and its definition will be the preliminary requirements for the system. Just as you do not need to develop the full requirements before you build the prototype, you also should not attempt to build the prototype without some kind of a definition of it.

Defining the prototype before building it helps to think through the basic functions of the system. It is a valuable thing to do, even if the prototype may change considerably when the user actually begins to see its outputs.

3.3.2 Stages

The stages in the RAD methodology is consist of planning, analysis, design, build/coding, test, and implementation.

Step1: Planning

Planning phase is the first phase of the PESAT application development process. It is also known as preliminary investigation phase. In this phase, the scope of the project and the perceived problems that triggered the project are defined. This planning phase is established by the survey phase in terms of scope, development strategy, schedule, and resource requirements.

This planning phase or preliminary investigation phase is concerned primarily with the view of the existing systems. The big picture, not details is concerned. Furthermore, whether resources will be committed to the project is determined.

There are a few tasks that need to be done in this phase for SAT development, which are:

1. Problem listing
2. Scope definition and negotiation
3. Scope definition and negotiation
4. Project worth assessing

Problem listing

One important task in this phase is to establish an initial baseline of the problems that triggered this project, such as how does the PESAT look like, how does it be implemented, etc. Besides that, the perceived constraints (limits) of the project such as durations of the certain tasks are listed also.

Scope definition and negotiation

The scope of PESAT application is defined and negotiated. It is to determine the full picture of the entire project. The PESAT application is divided into five major modules, which are System Security Module, Permission Security Module, Computer Security Module, Virus Scanner Security Module and Power Control Security Module.

Project worth assessing

This is to determine the worth of PESAT application to be developed. Several discussion and survey need to be gone through to determine the project is considered worthy and approved to continue.

The project planning

After the project is considered worthy to continue, the PESAT application is plan in depth. The planning includes schedule and resource requirements for the entire project, such as development tools, references, etc.

Step 2: Analysis

Analysis phase in the SAT development consists of problem analysis, requirements analysis, and decision analysis.

1. Problem Analysis
2. Requirement Analysis
3. Decision Analysis

Problem Analysis

There are a few current or existing systems in the World Wide Web. This problem analysis phase provides thorough understanding of the problems that triggered the project. Sometimes, it may be known as the study phase, study of the current system, or detailed investigation phase. It is concerned primarily with the views of the existing systems. It produces system improvement objectives that address the problems. Problem analysis includes the tasks of studying the problem domain, analyzing problems, and establishing system improvement objectives.

The problem facing by the similar current systems is how to place the working security application to work fully in multiple users operating system platform. The current existing systems are easily to disable by entering the operating system using another user.

Requirement Analysis

Requirement analysis phase defines the requirements for the PESAT application. The key in the phase is 'what'. In this phase, the users' needs and what they want from the PESAT application are determined. It is also known as definition phase or logical design phase. Requirement analysis includes the tasks of defining requirements, analyzing requirements, tracing and completing requirements, and also prioritizes requirements.

What the users really need and what they want from the PESAT application is a user-friendly application system. It must incorporate many security measures in the application, not using many hard jargons, apply the masseurs quickly and small in size.

Decision Analysis

Decision analysis phase identifies the solutions, analyzes those solutions, and recommends the PESAT system that is designed, built, and implemented. A system proposal is produced, which fulfills the requirements identified in the previous phase. Decision analysis phase includes the tasks of identifying solutions, analyzing solutions, comparing solutions, and finally recommending a solution.

The solution for the problems exist in the current systems is to develop a more advance application, where the PESAT application is recommended. Besides having the same functionalities as the current systems, the PESAT application will be able to protect the computer in the system tray, using low computer resources, alert user of computer security leakages and auto update the virus definition.

Step 3: Design

Design phase is defined as those tasks that focus on the specifications of a detailed solution for the PESAT application. It is different from analysis phase. Analysis phase emphasizes the problems, whereas design phase focuses on the technical or implementation concerns of the PESAT application. There are many popular strategies or techniques for performing systems design, such as Modern Structure Design, Information Engineering (IE), Prototyping, Joint Application Development (JAD), and Rapid Application Development (RAD). For the PESAT design, prototyping and RAD methodology are the strategies that have been used.

Step 4: Build/Coding

After the design phase, it comes to the build/coding phase. It is normally known as construction phase. The construction phase is the build/coding, installation, and testing of the system components. It develops and tests the functional system, which fulfills the design requirements and implements the interfaces for the PESAT application. Programming, implementation and integration of software components are generally recognized as major aspects here. The tasks that are included in this phase are installing the development tools or software packages, and write the programming code for the entire PESAT application.

Step 5: Test

Testing is one of the most important phases in the PESAT application development process. There are two levels of testing to be performed in the PESAT application, which are:

1. Stub Testing
2. Unit or Program Testing
3. System Testing

Stub testing

It is the test performed on individual events or modules of a program of the PESAT application. In other words, it is the testing of an isolated subset of a program.

Unit or program testing

It is a test whereby all the events and modules that have been coded and stub tested for a program are tested as an integrated unit; it is testing of an entire program of the PESAT application.

System testing

It ensures that application programs written and tested in isolation work properly when they are integrated into the PESAT application.

Step 6: Implementation

Implementation phase is the delivery of the PESAT application into day-to-day operation. This phase includes the tasks of conducting system test and preparing conversion plan.

1. Conducting system test
2. Preparing conversion plan

Conducting system test

A system test is conducted to determine whether the PESAT can operate correctly or not. The primary inputs to this task include the software packages, custom-built programs, and existing programs comprising the PESAT application. The system test is done using the system test data that was developed earlier. As with previous tests that were performed, the system test may result in required modifications to programs, thus, once again prompting the return to a construction phase task. This iteration would continue until a successful system test was experienced.

Preparing conversion plan

Once the successful system test has been completed, the PESAT application is prepared to be place into operation. By using the design specifications for the PESAT, a detailed conversion plan is developed. This plan identifies documentations that need to be developed.

The conversion plan also includes a system acceptance test plan. The system acceptance test is the final system test performed by end-users using real data. It is an extensive test that addresses three levels of acceptance testing, which are:

1. Verification testing, which runs the PESAT application a simulated environment using simulated data.
2. Validation testing, which runs the PESAT application in a live environment using real data. This is sometimes called beta testing.
3. Audit testing, which certifies that the PESAT application is free of errors and is ready to be placed into operation.

3.4 Requirement Elicitation

In order to learn more about the development application and determine the requirements of the user, numerous techniques are initiated in the PESAT application. These include research conducted in library, internet, document room and brainstorming.

Library Research

A lot of informative resources are available in the library. There are a wide range of journals, articles, reference books and conference listing in the library. The gathered information is the precise view point within the professional. It does help in the PESAT development by defining a lot of numerous security measures.

Internet Research

The information in the library may not the up-to-date resources. This draws to the research on the World Wide Web. There are a lot of resources in the internet. By simply typing the key words in the search engine, a list of the related information are displayed. The next job is to navigate through the listings, filter the desire information and learn from the information.

Document Room Research

The document room as the faulty facility does provide a research centre for the PESAT development. It has a lot of the past report of the final year students. PESAT development has used the methodology referenced in the document room. In short, the research and analysis in the documents room has been a great help to give a better view for the development of PESAT.

Brainstorming

During the process of requirement elicitation, brainstorming is often carried out. It generates ideas, whether good or bad is not a matter. Brainstorming is important as a lot of uncertainty is easily seen in the requirement elicitation. It normally conducted in the discussion, whether with the supervisor or colleagues. It discovers and improves the development of PESAT application.

3.5 Summary

In the application development of the Personal Security Administrative Tool, the iterative rapid application development methodology is very well deployed in the application development. It ensures all the stages are repeatedly reviewed and testing along the PESAT development and consequently reduce the likely errors or bugs come across the application. The application will be developed based on the stages in the iterative RAD. RAD accelerates the requirements analysis and design phases through an iterative construction approach. On the same time, it reduces the amount of time until the users begin to see a working system. Parts of the process are repeated as system process evolved. This is so-named as iteration process. Prototyping assure the system to meet the performance goals or constraints and fulfill the users' requirements.

CHAPTER 4

SYSTEM ANALYSIS

4.1 Introduction to System Analysis

System analysis in the application development is an essential and important phase. In term of software life cycle, system analysis is used to determine what a system does and analyzes the system needs. Typically, system analysis requires a few day times to complete for small scale project, and it might consume a few weeks time for large project, such as banking system, stock exchange system and airline system, depends on the complexity of the application.

In other perspective, system analysis can be a requirement that contains feature of the system or a description of something the system is capable of doing in order to fulfill the system's purpose. In further, a software requirement definition is an abstract description of the services, which the system should provide, and the constraints under which the system must operate. The development of PESAT application will include both requirement analysis, namely functional requirement and non-functional requirement.

System analysis phase in the context of Personal Security Administrative Tool, PESAT is to acquire general knowledge of how the application works. In such, an overall understanding of system data flow and system process in the PESAT development will be acquired. By then, system analysis will reveal on the how the available system of this type had been developed. A completely research will contribute to how the PESAT

application can be developed using the existing technology, or with the latest emerging technology in the market. Besides, system analysis will analyze the major application components, software and hardware requirement in order to develop a robust and reliable PESAT application.

4.2 Functional Requirement

Functional requirement is a statement of the service or functions that a system should provide how the system reacts to particular inputs, and how the system should behave in particular situations. In some cases, it also stated what the system should not do. Furthermore, it is independent from the implementation of the solution. There are five modules recognized as the most important functional requirements in the PESAT development, namely System Security Module, Permission Security Module, Computer Security Module, Virus Scanner Security Module and Power Control Security Module. The respective module contains many more sub-modules, as describe below.

System Security Module

The System Security Module is the security measures to be imposed in the PESAT application. There are two sub-modules in this component, namely Password Management Sub-module and Application Start-up Sub-module.

1. Password Management Module has the functionality to protect the application authorization using password. Only the authenticate user with a valid password can proceed to the other parts of the PESAT; likewise, the application automatically quit after three consequent password failure. This module also contains a password generator to assist the user to build a strong password.
2. Application Start-up Module has the functionality to provide the user choices whether to automatically start-up the PESAT as Windows loading, or by manually start-up the application; and whether to hide the application in the system tray or not.

Permission Security Module

The Permission Security Module is consisting of File Folder Locker Sub-module, Application Restriction Management Sub-module, Program Aliases Management Sub-module and Add Remove Program Manager Sub-module.

1. File/Folder Locker Module is to provide password to protect the file or folder. This module assists the user to protect their sensitive information from being accessed by the intruder.
2. Application Restriction Management Module is to restrict or allow the application to be running in the system. For instance, some user might restrict the other users to use the Word application, so using this module, other users is not allowed to launch the Word application.
3. Program Aliases Management Module is to match, locate or relocate the correct path to the application installed in the computer. This functionality is useful if the path to the correct application location is biased.
4. Add/Remove Program Module is to simplify the task to add program into the system or remove the installed program within the system. This is similar to the Add/Remove Program in the Windows Control Panel.

Computer Security Module

Computer Security Module is dealing the setting to the computer system. It has several sub-modules like General Computer Management Sub-module, Drive Visibility Management Sub-module, Port Visibility Management Sub-module, Desktop Management Sub-module and Internet Management Sub-module.

1. General Computer Management Module is to totally disable the Window Control Panel, or hide a part of the Control Panel applet; it can disable the registry editor, REGEDIT from the user; it can also disable CD-Rom auto run feature; This module can customize the Start Menu item, like remove the RUN, FIND, SHUTDOWN, etc in the menu and alter menu show delay duration so that the

menu will show up faster. It can also control the right mouse click, whether to disable the right mouse menu or customize it.

2. Drive Visibility Management Module is tending to hide or show the desired drives to the user. Using this module, the user can control the use of floppy or CD-Rom Drive to other user effectively.

3. Port Visibility Management Module is tending to hide or show the desired communication port, COM port or printer port or LPT port to the user. Using this management, the user can control the use of modem and printer to other user effectively.

4. Desktop Management Module is to lock the desktop completely, so that the user is not able to change any desktop setting; it provides a password-protected screen saver, so that the unauthorized user is not able to access the computer easily.

5. Internet Management Module is designed for user to delete cookies, temporary files, history, IE-typed history and recent document histories.

Virus Scanner Security Module

This module will keep the antivirus of the computer system up-to-date. It has 2 sub-modules, namely Update Notification Alert Sub-module and Update Automation Sub-module. It is built as the combination with the Symantec Norton Antivirus 2003. Connection to the Internet is required in this security module.

1. Update Notification Alert Sub-module is to prompt user about the last date virus definition being updated and whether new patch is being released from the antivirus vendor. It alerts the user of updating the virus definition if available.

2. Update Automation Tool Sub-module is to provide automatic or manual virus definition update. It connects to the Symantec, check the available virus definition and download the packages to the host computer.

Power Control Security Module

This Module is consisting of two sub-modules, namely Service Manager Sub-module and Shutdown Manager Sub-module.

1. Service Manager Sub-module is a monitoring tool to check the service currently running in the Windows; and able to detect if there is any non-responding application, waiting for the user command to terminate it. Using this module, PESAT assures that the operating system would be stabile all the while.
2. Shutdown Manager Sub-module contains a few shortcut commands to quickly shutdown, logoff and restarts the computer without using the conventional way. It is the shortcut to those commands.

In short, System Security covers PESAT password authentication mechanism and application auto or manual start-up; Computer Security controls drive or port visibility; desktop, internet and mouse management; Power Control Security provides one-key to shutdown, log-off and restart computer; Permission Security shields files or folders with password; restrict user access to certain programs and re-aliases the programs' location; Virus Scanner Security gives the user optional whether to auto or manual update the virus definition from the Internet.

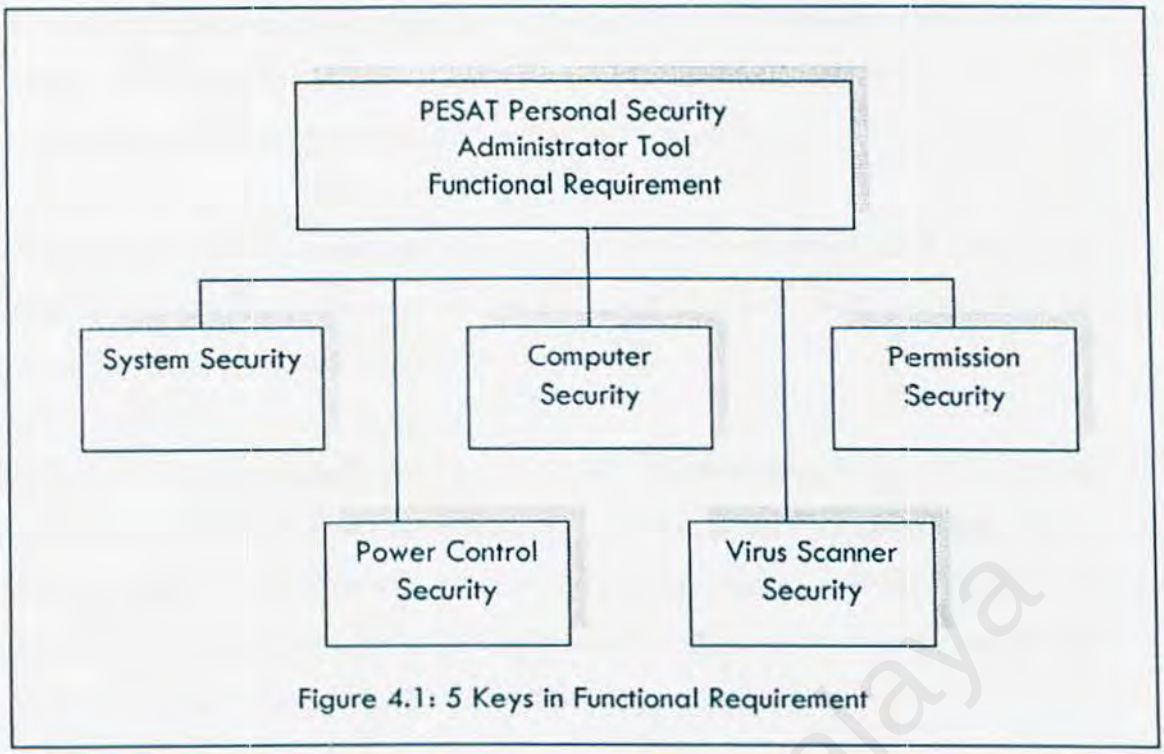


Figure 4.1: 5 Keys in Functional Requirement

4.3 Non-Functional Requirement

Non-functional requirements are the constraints on the service of function offered by the system, in which the system must operate on the standards to meet the delivered system. They include the system timing constraints, constraints on the development process or standards such as choice for constructing a solution to the problem. The PESAT application must assure the non-functional quality such as user-friendliness, robustness, modularity; reliability, security, expandability as well as maintainability are included in the development.

User-Friendly Interface

The PESAT application would have an attractive and user-friendly interface. This help in reducing the learning curves, where the new user may drive the application in ease. User-friendly interface should take consideration in user familiarity and consistency, whilst the implementation must include the meaningful and intuitive icon and menu.

The user interface must place the user in control, where the interaction modes will always be the user desired actions, with the flexible communication between the application via mouse movement and keyboard commands.

The interface must take adequate consideration in user's memory load. This principle intends to reduce the demand on short-term memory, by decline the requirements to remember past actions and results.

The interface design should apply to consistent fashion where all visual information must be organized according to a design standard that is maintained throughout all screen displays. Apart from that, input mechanisms are constrained to a limited set that are used consistently throughout the application. Lastly, mechanisms for navigating from task to task are consistently defined and implemented.

Robustness

All the modules in the PESAT application will be wholly tested in the system testing phase, where any detected error will be solved immediately. The measures provide quality of robustness to meet the system expectations, avoid unnecessary disaster and reduce the possibility of failures during the implementation of the system.

Modularity

The system shall be developed with a modular approach to ease maintenance and scalability of any modules in the system. This remarks that the application can be modified and enhanced easily in the future. The modular approach will reduce the development time and prevent redundancy in the codes.

Reliability

The developed application must be reliable in overall performance. The PESAT application is said to safe guard the computer, and should not cause unnecessary downtime of the overall performance. In further, it must be easily to be maintained simplicity and effective. It must run smoothly in the background environment, without

turning over the overall operating system performance. The application should not produce dangerous or costly failures when it is used in a reasonable manner.

Maintainability

Maintainability is the ease with which a program can be corrected if an error is encountered, adapted if its environment changes, or enhanced if the customer desires a change in requirements. If the application is developed not using the good programming practices, the maintainability of the application would be difficult. Process of maintainability is strongly to be implemented in the PESAT application to fix out the bugs or system faults in the shortest time.

Security

The PESAT application has a password-log-in interface to allow the only authorized user to conduct the application. The security measure is to minimize the risk of data exposure to unauthorized people. A lot have to be taken in the development of the PESAT application.

Response Time

The desired information or functionality ought to be available instantly to the user at any point of time. The user should not be kept waiting for a long time.

Expandability

In future, the PESAT application is able to enhance and add-in new features, functionality, and volumes of product. This is because the current development of the PESAT application is specially designed to leave some areas to be developed in the future time.

In short, PESAT assures eight of non-functional requirement to be implemented in the development, namely a user-friendly interface; modularity to reduce develop-time, maintainability to correct errors; very fast in response-time, robustness to avoid from

unnecessary failures; reliability in overall performance; securable from being accessed by unauthorized user; and ability to be expand in the future.

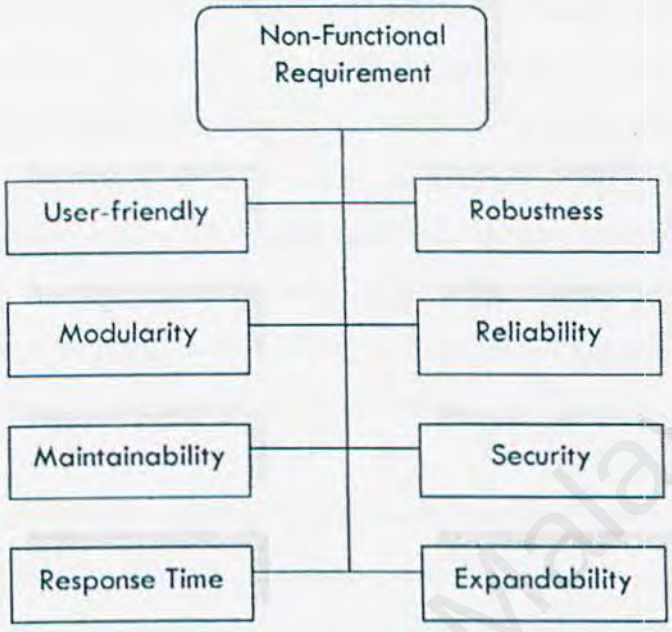


Figure 4.2: 8 Keys in Non-Functional Requirement

4.4 Technology Consideration

The PESAT application is developed based on the most current technology research. It is an application developed using the most well known programming language, namely Microsoft Visual Basic 6 and the Windows Script Host, WSH; with the assist of bundle of software like Adobe Photoshop 6 and other tools. PESAT is developed to be used in Windows XP Professional Edition. Mainly, Visual Basic 6 programming language is used to program the overall system, whereas WSH is used to program the Windows XP core objects in the application, and Photoshop is deployed to beautify the application interface.

4.4.1 Operating System Platform

Operating System Platform Chosen: Windows XP Professional

The PESAT application is developed best for Windows XP Professional. This version of the operating system is well-known to the world and mostly being deployed in all around the corner. Compare to Windows 2000 which is old brand operating system, and the old version of Windows like ME, 98 and 95, Windows XP Professional is believed the best operating system across the century that embarks with its stability and reliability. Besides, the professional edition also provides essential networking technology that can be implemented in the networking environment. This is why PESAT application is developed for Windows XP Professional.

In general, Microsoft's Windows XP is built to work with a series of microprocessors from the Intel Corporation that share the same or similar sets of instructions. Integrating the PESAT application in Windows XP would make the operating system gaining more security measures.

Technically, Windows XP provides a bundle of benefits. This draws the consideration of why PESAT manipulates Windows XP Professional as its core compatible operating system. It contains the below strengths:

1. Performs primary functionality and maintain stability.
2. A very reliable operating system.
3. Being a mostly deployed operating system for personal as well as work station.
4. Contains a lot of hidden functionalities; PESAT is developed to unhide this hidden functionalities.
5. Do not read from or write to Win.ini, System.ini, Autoexec.bat or Config.sys on any Windows operating system based on NT technology.
6. Perform Windows version checking correctly.

4.4.2 Application Programming Language and Development Tool

Windows Script Host

Windows Script Host (WSH) was designed to provide a system-wide environment that helps automate tasks across all 32-bit versions of the Windows operating system. WSH is a part of microcosm in Windows called Windows Script, within which a number of Microsoft technologies are conveyed, including the Windows Script Engine (also known as the ActiveX Scripting engine), COM Automation, VB-Script, J-Script, and the Windows Script Components (WSC), formerly called script-let. WSH is a component of Windows 98 and Windows 2000, and is also available as a separate add-on for Windows 95 and Windows NT 4.0.

Until WSH came along, there was no powerful and versatile way in Windows to execute batch code to automate repetitive and administrative tasks such as login, file management, setup/uninstall, registry tweaks, shortcuts, creation of user accounts, and so forth. Creating a shortcut is quick and easy, but it is much harder to create 100 shortcuts at a time. Script languages such as VBScript, JScript, or Perl, are far easier to use than C++ or even Visual Basic. In an everyday work, we probably have lots of tasks that we can't easily accomplish manually, but aren't worthy of even the smallest Visual C++ based application. WSH fits this niche and, as a result, has earned its well-deserved success.

WSH contains the below listings of advantages:

1. Introduction of a new, XML-based file format to describe the scripts to execute
2. Ability to write and run programs that are stored as text, which makes those programs exceptionally easy to write and distribute
3. Introduction to the ability to sign scripts and to set policies about running scripts
4. Take advantage of the extensive code-security capabilities that Windows platforms provide

Microsoft Visual Basic 6 Programming Language

PESAT application development is using visual basic programming language to program the overall module. Visual Basic is very suitable to build the PESAT application as PESAT is a simple security tool correlate to the core Windows operating system. In this case, Visual Basic plays a remarkable role in the PESAT development. Besides, it is fast in term of execution as the programming languages is specially designed to fit in the Windows environment, compare to JAVA programming language.

Visual Basic programming language has the ability to create the graphical user interface easily. Rather than writing numerous lines of code to describe the appearance and location of interface elements, development of PESAT simply add pre-built objects into place on screen. Most of the skills necessary to create an effective user interface is easy and novice.

Visual Basic programming languages provides the below listings of advantages to PESAT:

1. It is the fastest and easy way to create application for Microsoft Windows, no matter the programmers are an experienced professional or brand new to Windows programming.
2. Visual Basic provides the development with a complete set of tools to simplify rapid application development. It includes all intrinsic controls, plus grid, tab, and data-bound controls.
3. The Active-X technologies in the Visual Basic is suitable for PESAT development as it automates applications and objects with the functionality provided by other Microsoft's product.
4. The finished application is a true .exe file that uses a Visual Basic Virtual Machine that can freely distribute.
5. The PESAT application built by Visual Basic programming language is expected to be extremely fast in run-time and smoothly in the Windows application.
6. A very well-constructed online documentation and MSDN on the internet to solute most of the PESAT application development difficulties.

7. Windows Script to be written is using Visual Basic Script, VB script approach. So PESAT application is to be Visual Basic programmed.

Symantec Norton Antivirus 2003

One major functional in the PESAT application is the virus scanner security module. In this module, PESAT is said to have the ability to manually or automatically update the antivirus definition to the computer system. In term of auto update the virus definition, the PESAT application must be able to determine the last date of the virus definition, and go online to download the latest virus definition. This can be done by checking the virus definition in the file "VirusDefs" in path "Common Files\Symantec Shared"

Norton Antivirus provides comprehensive virus prevention, detection, and elimination software for your computer. It automatically finds and repairs infected files to keep your data secure. Easy updating of the virus definition service over the Internet keeps Norton Antivirus prepared for the latest threats. Worm Blocking and Script Blocking increase protection by detecting new threats before virus definitions are created.

Adobe Photoshop 6 Graphical Editor

The PESAT application development is not solely depends on the programmatic language. In such, when dealing in the graphical interface design, PESAT uses Adobe Photoshop 6 as the graphic editor. This development tool is used to design, draw and re-edit the graphical image or picture placing in the PESAT application. It contains much excellent functionality as the graphical editor. Below are some of the advantages:

a) Working with vivid, luxurious and invigorating color
b) Transforming, re-touching and re-position the image professionally
c) Drawing and editing the image in ease
d) Contains filters and special effects
e) Many pre-defined automating task simplify the task

Batch File Compiler 5.1

The application is used to compile binary code written in the Windows Script Host. This will create the execution of the scripting language.

WMI Tool Set

The application tool set comprises CIM Common Interface model Studio, Event Registration, Event Viewer and Object Browser. The utility is a simple application to sketch the windows script, locate the script in the system and execute the script to the system. It contains numerous advantages as a tool set to control Windows Management Instrumentation, WMI in short.

1. WMI Object Browser can be used to view objects in a Common Information Model (CIM) repository. It can also edit property values, edit object and property qualifiers, and run selected methods.
2. WMI CIM Studio can be used to view classes in a CIM repository. It can also edit classes, properties, qualifiers, and instances; run selected methods; generate and compile Managed Object Format (MOF) files; and generate framework provider code.
3. WMI Event Registration Tool is a graphical interface used to configure permanent event consumers. Using the tool, users can create or view instances of event consumers, filters, bindings, and timer system classes.
4. WMI Event Viewer displays events for all instances of registered consumers. It can also start WMI Event Registration Tool from WMI Event Viewer.

Conclusion

PESAT is applicable in Windows XP Professional Operating System. Windows XP is chosen of its reliability and stability in overall performance compare to other OS in Windows families. PESAT is constructed using Visual Basic programming language and Windows Script Host Language. This is suitable as PESAT is a Windows-base application. Visual Basic has the advantages of easy interface design, true execution

application and fast in run-time. PESAT uses Adobe Photoshop to design the graphical interface. PESAT uses Symantec Norton Antivirus common file in building Virus Scanner Security Module.

Operating System Platform	Microsoft Windows XP Professional
Programming Languages	Visual Basic, Windows Script Host
Development Tool	Microsoft Visual Studio Basic 6, Adobe Photoshop 6, Symantec Norton Antivirus 2003, Batch File Compiler 5.1, WMI Tool Set

Table 4.1: Technology In Use

4.5 System Requirement

The system requirement is consist of hardware and software requirement. In the PESAT application, the system requirement for respective developer and user is indifferent. The developer's system requirement is basically higher than the user's system requirement.

Developer Requirement

Hardware	Requirement
Processor	Pentium 3 800 MHZ and above
Memory	Minimum of 256 SDRAM
Hard Disk Space	Minimum of 20 GB space
Other	Modem Card and other standard Computer Peripherals

Table 4.2: The Developer Hardware Requirement

Software	Requirement
Operating System	Windows XP Professional
Programming Tool	MS Visual Studio 6
Other	Adobe Photoshop 6

Table 4.3: The User Software Requirement

User Requirement

Hardware	Requirement
Processor	Pentium 2 400 MHZ and above
Memory	Minimum of 64 SDRAM
Hard Disk Space	Minimum of 5 MB space
Other	Modem Card and other standard Computer Peripherals

Table 4.4: The User Hardware Requirement

Software	Requirement
Operating System	Windows XP Professional

Table 4.5: The User Software Requirement

4.6 Summary

System analysis plays an important role in the development of the PESAT application. It is used to define and determine what a system really does and analyze the system needs. This is implemented from the functional requirement, non-functional requirement, technology consideration and system requirement. The frequent reviewing of the system analysis will assist the developer in gaining the advantages and knowledge about the implementation of the PESAT application. The following chapter will discuss about the design of the system, which includes the system functional design, description of system structure, description of process design and user interface design.

CHAPTER 5
SYSTEM DESIGN

5.1 Introduction to System Design

System design is a phase of Rapid Application Development where the entire requirement being produced from the system analysis for the system is translated into a representation of the system. This phase will focus on system functionality design, and user interface design, where system structure chart and data flow diagram are included in the system functionality design.

5.2 System Functionality Design

Large system can be decomposed into sub-systems that supply numerous related set of services. This is so called the system functionality design. This is an initial design process of identify and structure the significant sub-system into an independent software unit and establishing a framework for sub-system control and communication, where the respective sub-system sum up the entire system, by identifying and documenting their relationships.

System functionality design on the basis will decompose a system, using top-down approach, into a set of interacting sub-system. The design of the PESAT application is based on data flow oriented design. It is also called structured design. This type of design stresses on modularity, top-down design and structured programming.

System functionality design can be subdivided into context diagram, system structure design and system process design respectively, where context diagram, diagram 0 and DFD diagram 1 are packed in the system process design.

5.2.1 System Structure Design

The system structure of PESAT application is decomposed into five major components, which are the System Security Module, Permission Security Module, Computer Security Module, Virus Scanner Security Module and Power Control Security Module. Each major component is further divided into sub-modules. System structure design is a process oriented for breaking up a large program into hierarchy of structure chart of modules that result in a computer program, which is easier to implement and maintain. Structure chart is tree-like diagram. Components are factored, from top down, into modules. Studying the flow of data through the program derives the structure chart. Below shows the structure charts for the five major components.

a) System Security Module Structure Chart
b) Permission Security Module Structure Chart
c) Computer Security Module Structure Chart
d) Power Control Security Module Structure Chart
e) Virus Scanner Security Module Structure Chart

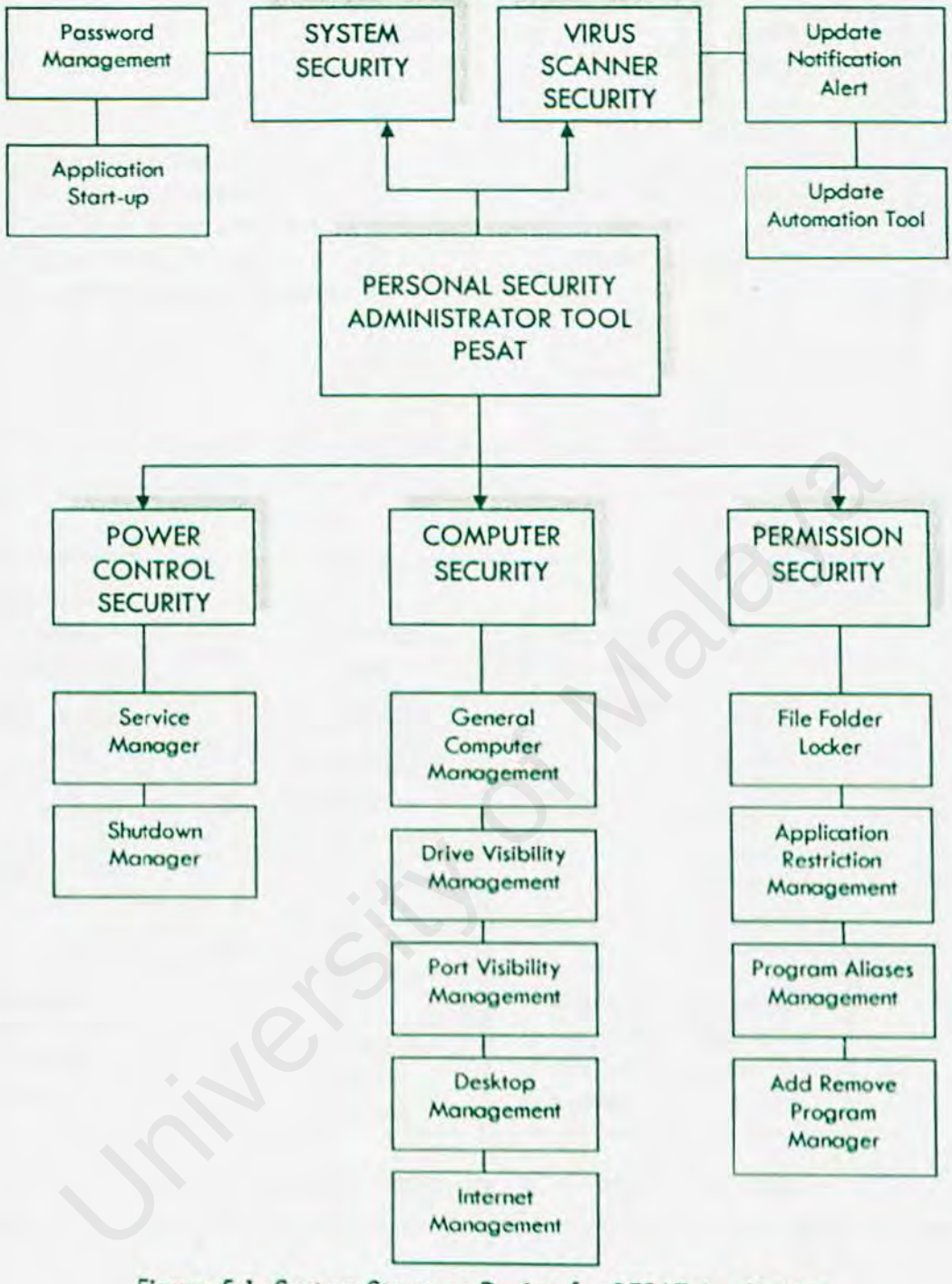


Figure 5.1: System Structure Design for PESAT Application

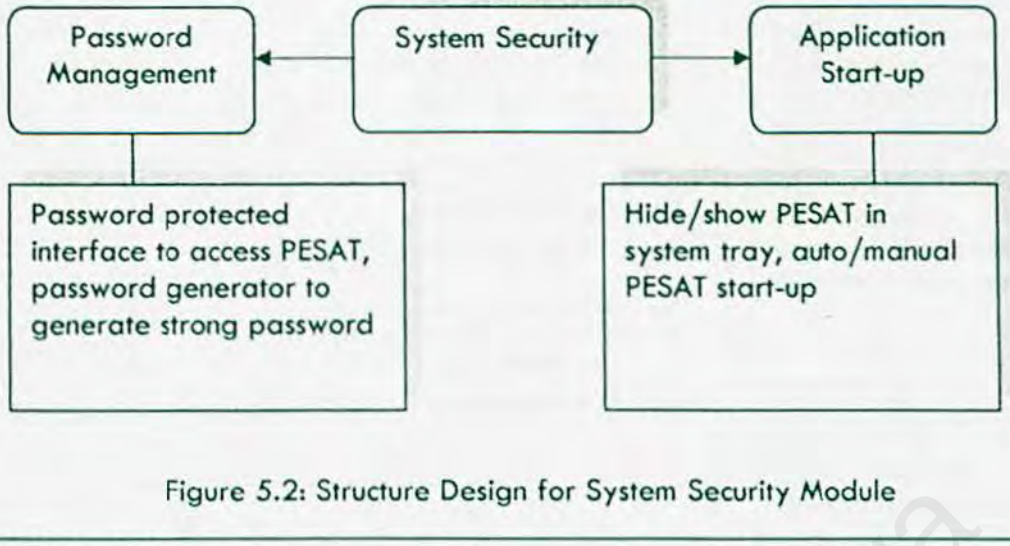


Figure 5.2: Structure Design for System Security Module

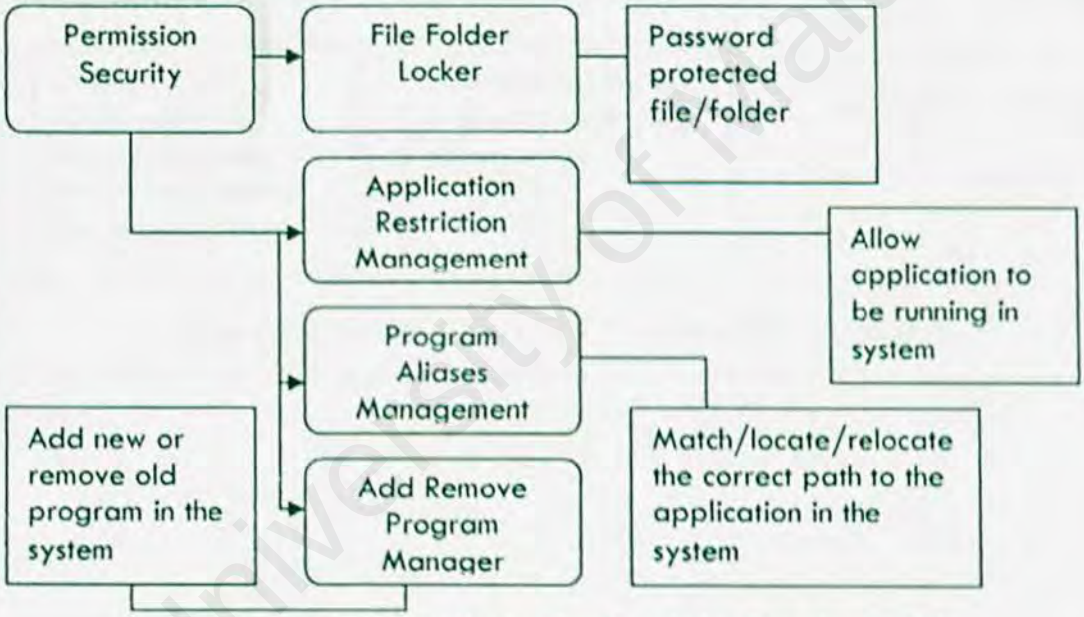


Figure 5.3: Structure Design for Permission Security Module

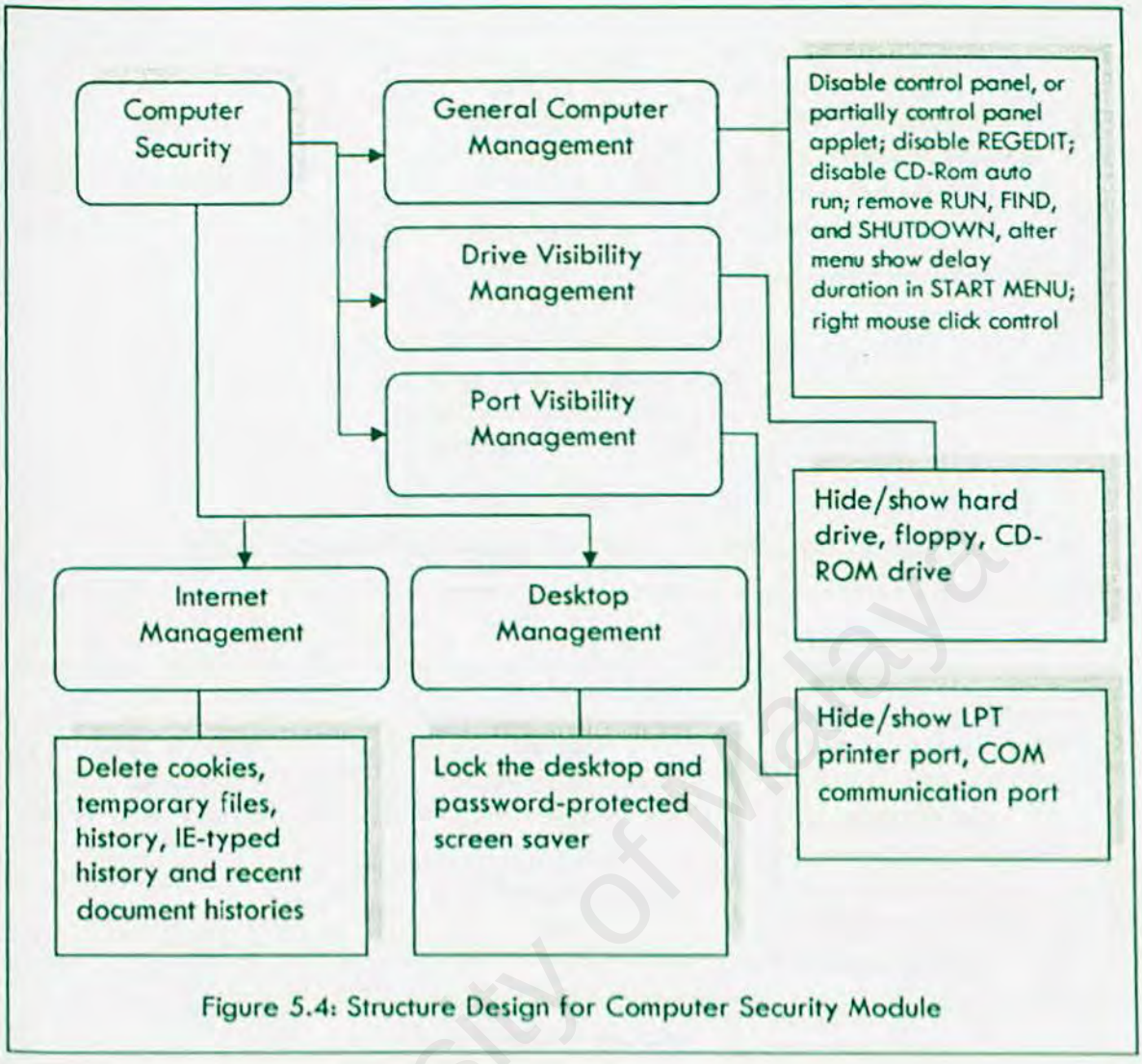


Figure 5.4: Structure Design for Computer Security Module

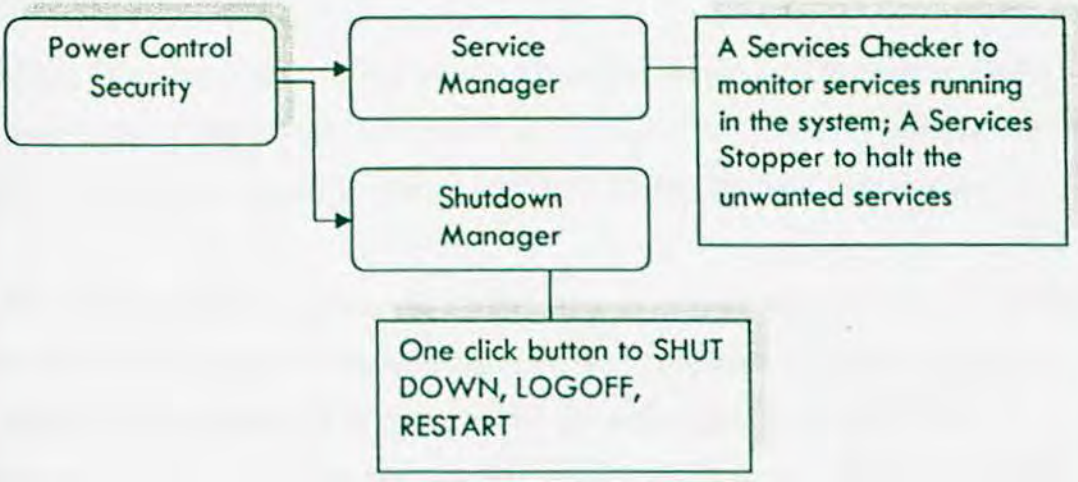


Figure 5.5: Structure Design for Power Control Security Module

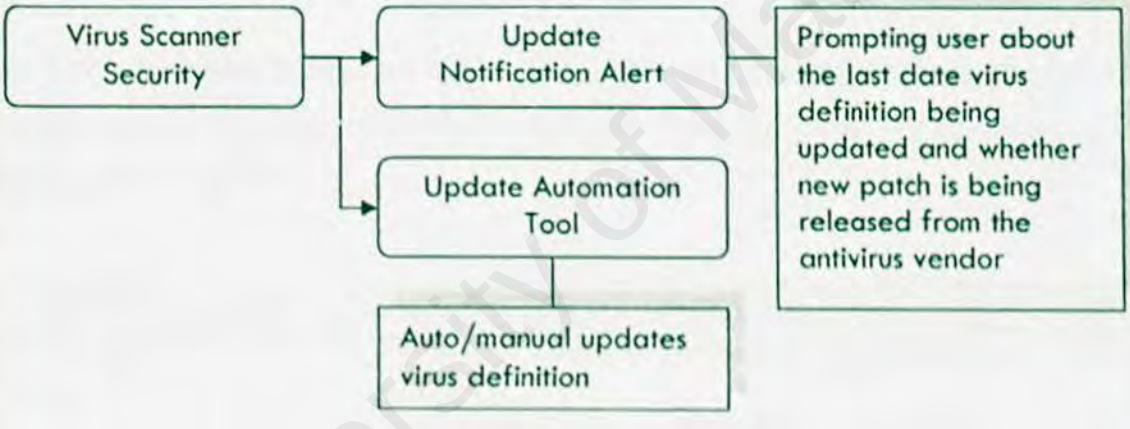


Figure 5.6: Structure Design for Virus Scanner Security Module

5.2.2 System Process Design

Data Flow Diagrams (DFD) is a process modeling technique used to show graphical characterization of the data process and flows in the system. The DFD gives an overview of system inputs and outputs, processes and flows of data through each process.

The Data Flow Diagram is extremely simple, ease in use and easy to learn. It provides an excellent of conceptual understanding of existing problems. It represents each flow of functionality in details and is the essential for the understanding of the PESAT application. It supports the decomposition using hierarchical approach. A Data Flow Diagrams deliberately suppresses the internal details of the transformations in order to focus on the architecture of the system as a whole. It emphasizes decomposition, components and interfaces.

Data Flow Diagrams usually are made after a Context Diagram has been created. The Context Diagram functions as the basis of a Data Flow Diagram. The following is the basic symbols of a DFD.

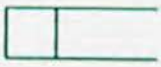
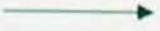
Symbol	Definition
	Transformation of data into another data
	Sources and destination of data
	Data in static storage
	Data on the move

Table 5.1: Data Flow Diagram Objects

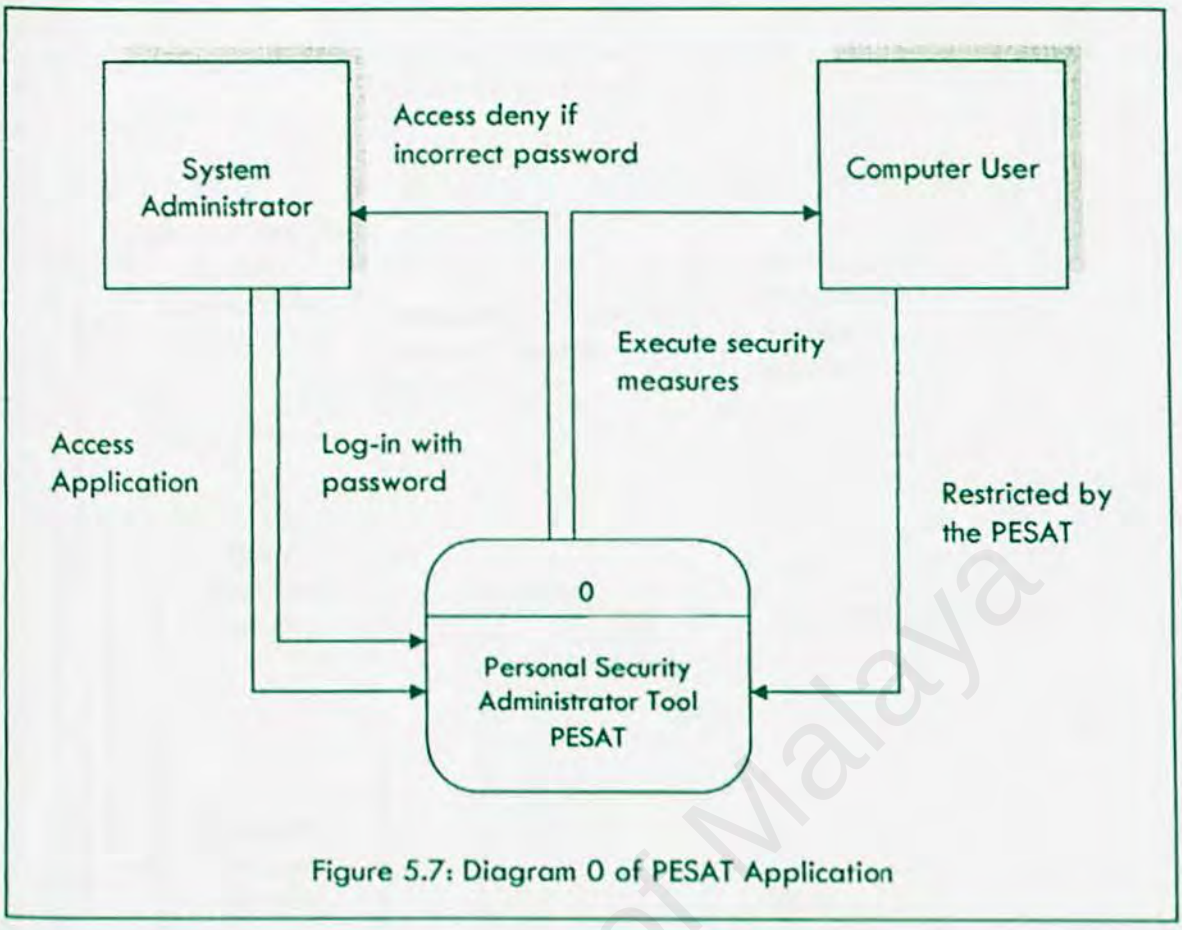


Figure 5.7: Diagram 0 of PESAT Application

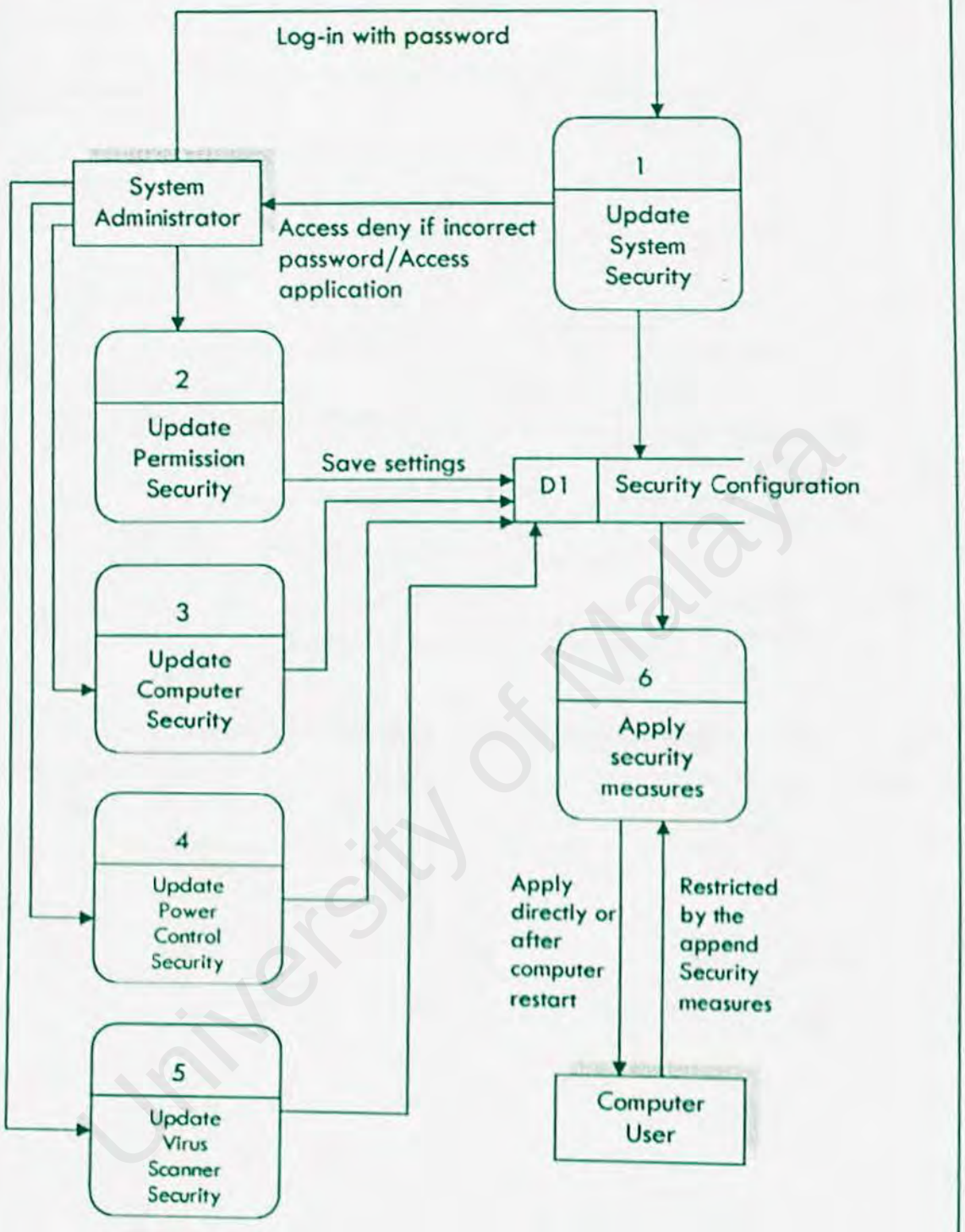


Figure 5.8: Diagram 1 of PESAT Application

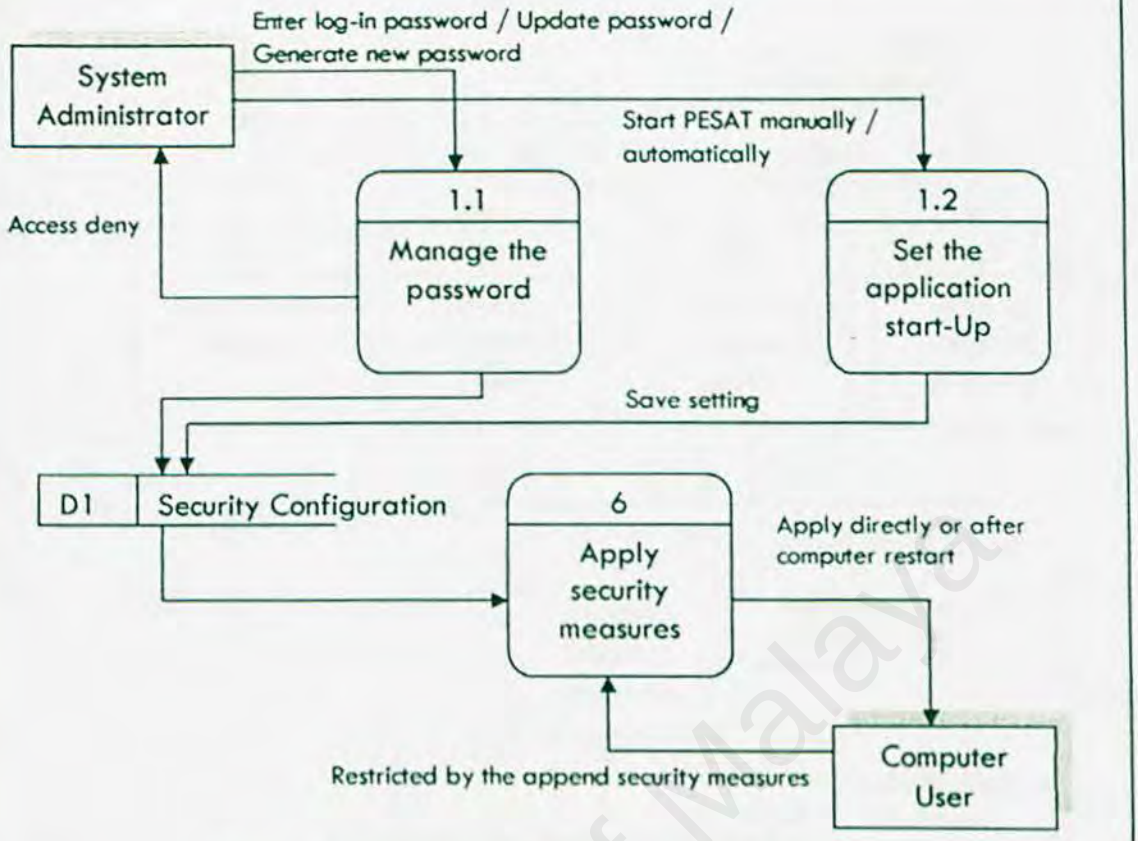


Figure 5.9: Diagram 1 of System Security Module

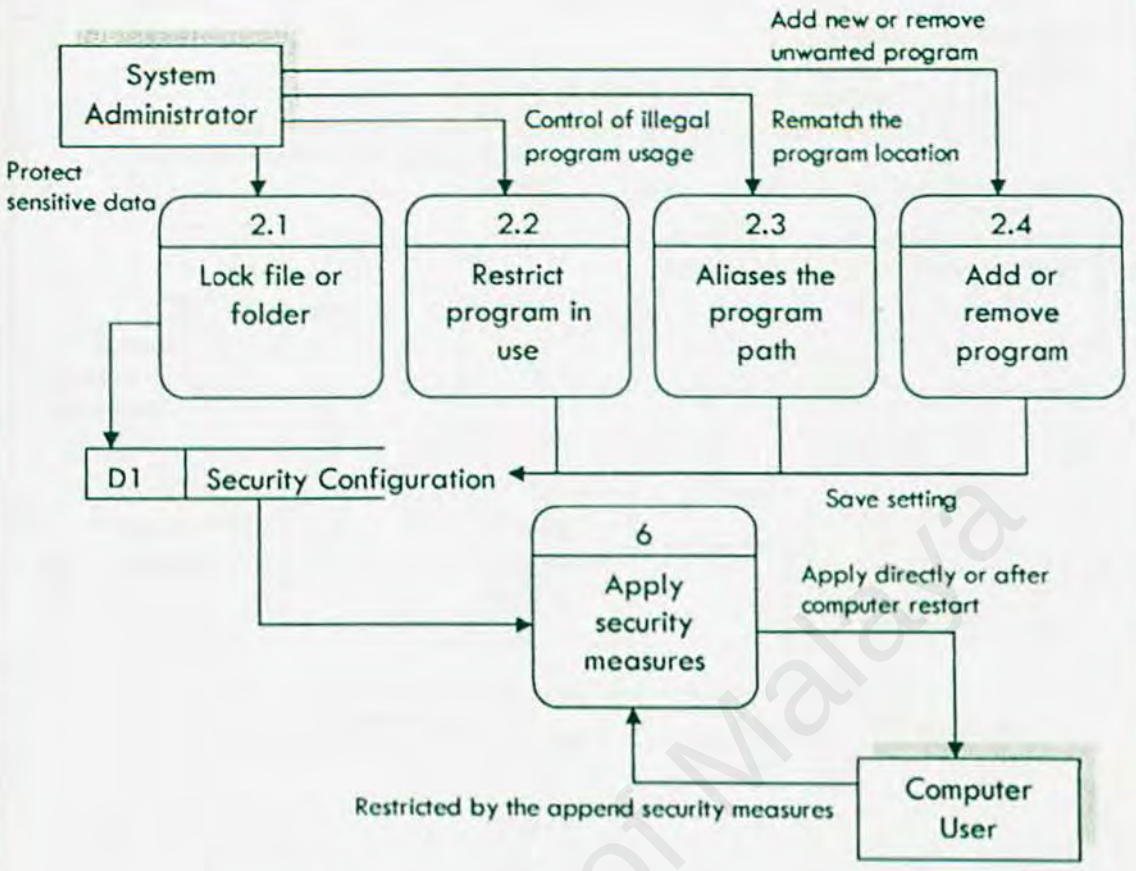


Figure 5.10: Diagram 1 of Permission Security Module

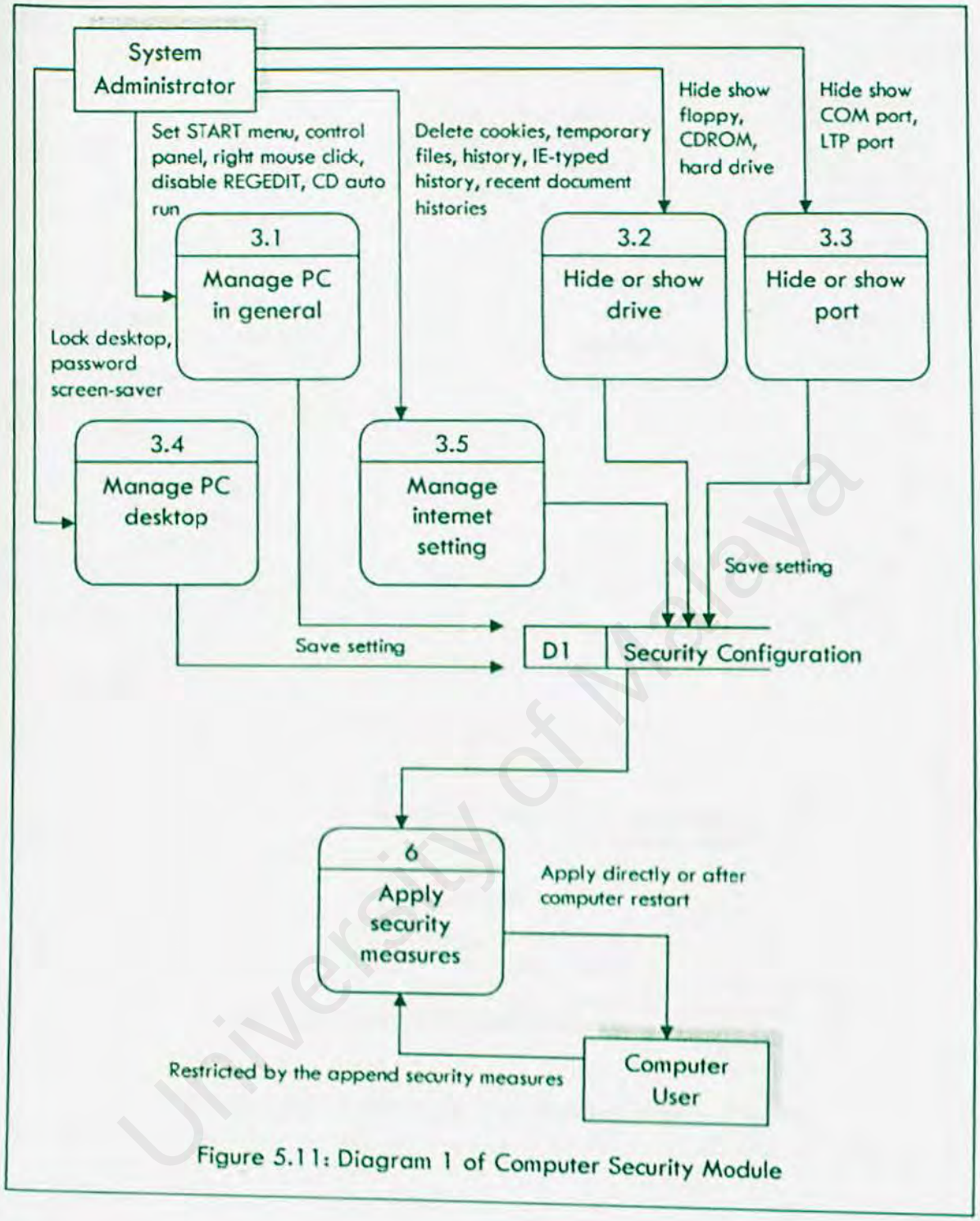


Figure 5.11: Diagram 1 of Computer Security Module

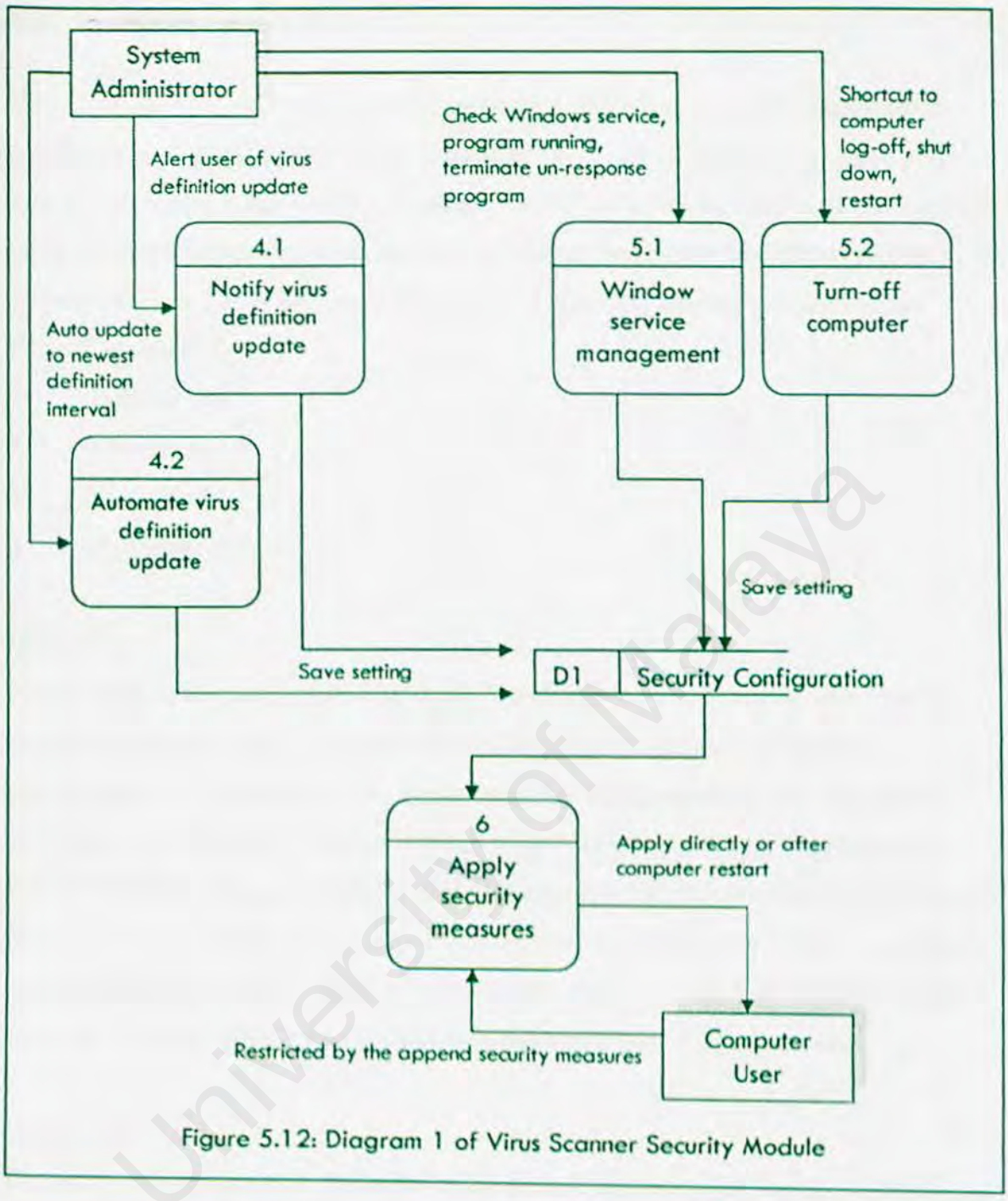


Figure 5.12: Diagram 1 of Virus Scanner Security Module

5.3 User Interface Design

User Interface Design is on of the most important element of the PESAT application development. Generally, PESAT must package itself into an attractive interface with ease of use, so that it can compete with the similar product in the market. A product with many excellent functionality but hard to use will not be able to stand steadily in the software market. The following is some general measures dealing in designing the PESAT application.

1. Ease of Use
2. Consistency
3. Navigation
4. Performance Issue

Ease of Use

PESAT application manipulates the color completely to provide an adorable sight smoothing the eyes. It uses the ocean blue as the background color to make the application more professional-like. It places its lots-of-functionality into the square boxes and round boxes so that the user can select them easily. PESAT application is built by skeleton design. A skeleton design means that the user can navigate through the major functions in ease. This followed by numerous functionalities relate to the security measures which is located neatly in boxes in side each skeleton. The interface design is very user-friendly, yet a resourceful of help documentation is also provided.

Consistency

Consistency brings a sense of identity to the PESAT application. User can easily locate and apply the appropriate security measures easily as the result of the consistency. All of the functionalities have been well categorized and place in the skeleton design respectively.

Navigation

The hierarchical organization and skeleton layout of the application would allow for simple and intuitive navigation between each component. It prevents a narrow and deep structure as it tends to get the user lost and off track. The user can easily navigate function by function in ease across the skeleton design.

Performance Issues

In term of performance issues, the PESAT application only packages the graphics which is considered important to the user. Too many graphics in the application would use the computer resources dramatically and reduce the performance indirectly.

The following figures illustrate the major user interface design in the PESAT application.

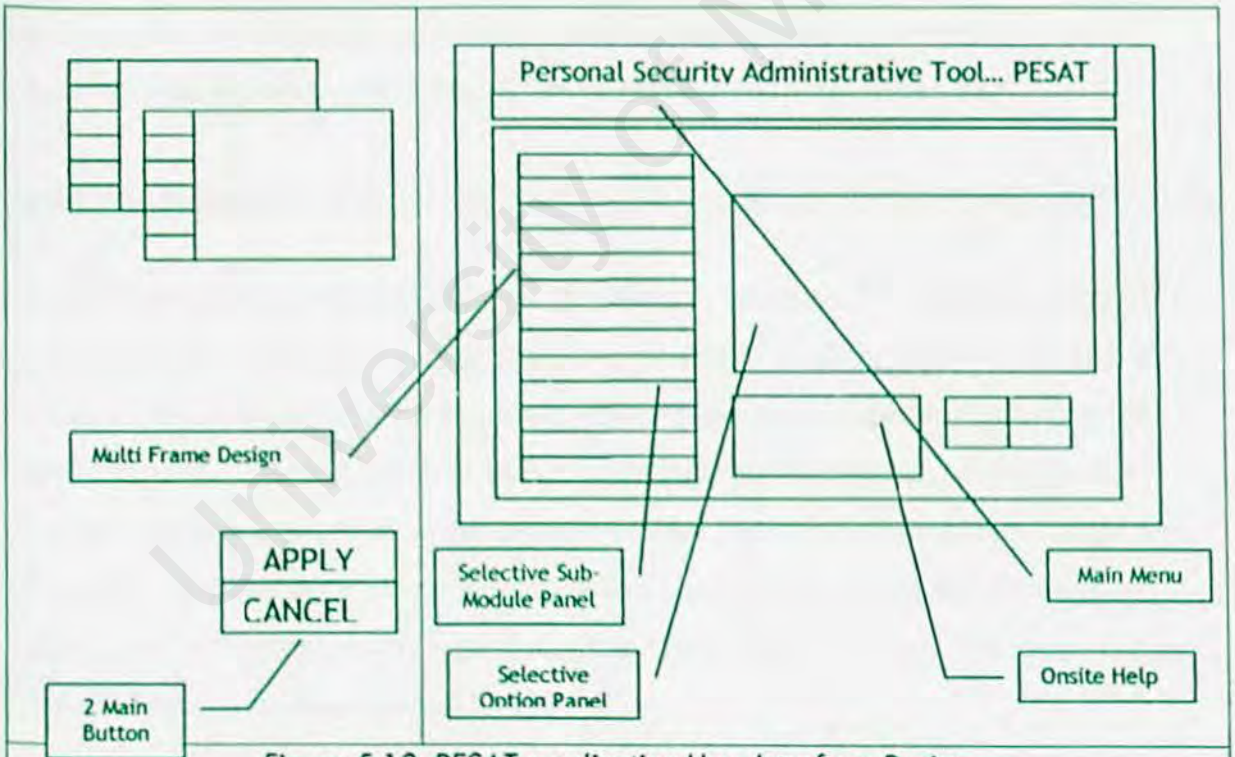


Figure 5.13: PESAT application User Interface Design

This is the PESAT application pre-plan user interface design. It is outclassed; authentic and classical interface that using user-friendly approach. It uses skeleton approach to fit in the five major modules namely System Security Module, Permission Security Module, Computer Security Module, Virus Scanner Security Module and Power Control Security Module.

Each module contains sub-modules that are placed in the left-hand side in the design, where selective options are located in the selective option panel. There is a useful onsite help which provides basic description of each option being selected. The main menu has some basic menu command in the menu bar like FILE, EDIT and VIEW.

In short, PESAT uses skeleton design to place the major functional. The PESAT title bar is banner-looking on top of the application design. It contains 3-D floating buttons and graphical design with vivid color. Left-panel contains major sub-module functionalities with more optional settings in the right panel. All the optional functionalities are well-describe with the inline status help. It can be minimized in the system tray.

5.4 Summary

System design is an important phase in the Rapid Application Development. It should be carefully implemented in order to get the overall system flows and to show clearly the ideas on how a system is to be developed. System design will determine whether the application will meet with success or not. It contains components and elements of a system and their appearance to the user. Functional and non-functional requirements in the system analysis stage are turned into design specification. From that the expected outcome of the system can be view through the user interface design. The phase is then followed by system implementation.

CHAPTER 6
SYSTEM IMPLEMENTATION

6.1 Introduction To System Implementation

System implementation is a phase integrating the designed modules or functions to develop a system based on the given requirements. It is the process takes place after the system design phase. This phase describes how the initial and revised process design put into the real work. Therefore, system development, coding methodology and development tools are included in this phase.

6.2 Development Environment

The development environment of PESAT application consists of software and hardware configuration. Using the suitable hardware and software will help in speed up the system development. The hardware and software tools that used to develop and document the system will be discussed as below.

6.2.1 Hardware Requirement

The hardware configurations used for developing the system are:

A Pentium 3 800Mhz Processor, with

- 256 MB memory
- 20 GB of free hard disk space
- 1.44 MB floppy drive
- 52X CD-ROM drive
- Other standard computer peripherals

6.2.1 Software Configuration

The software tools that have been used to develop PESAT Application are:

Software	Usage	Description
MS Windows XP Pro	System Requirement	Computer Operating System
MS Internet Explorer 6	System Requirement	Web Browser
Microsoft Visual Basic 6	System Development	Core Programming Compiler
MS Batch Compiler 2.4	System Development	WSH Programming Compiler
Adobe Photoshop 6.0	System Development	Graphic Editing
Jasc Paint Shop Pro	System Development	Graphic Editing
Microsoft Word	System Development	Project Documentation.

Table 6.1 Software Tools

6.3 System Development And Implementation

PESAT application is coded with the advancement of Visual Basic Language, and some integration script for instance, Windows Script Host.

6.3.1 PESAT Application Architecture

PESAT application is using a single form as the background layout. A lot of frames exist to form many cluster of functionalities. The frames that in the PESAT application is listed as below:

1. About PESAT
2. Password Management
3. Remove Orphanage Entry
4. Quick Access Panel
5. Drive Security
6. Folder Security
7. Application Restriction
8. Application Aliases
9. Start Menu Tweak
10. Desktop Tweak
11. Miscellaneous Tweak
12. Virus Scanner Intelli
13. Power Management

There are five modules inside the PESAT application, namely

1. Antivirus Module, correlating to the Norton Antivirus 2003
2. Core Module, the declaration of all the Windows API
3. Function Module, description of most PESAT application function
4. Password Module, correlating with the application password management
5. Registry Module, correlating with all of the registry management

There are two class modules inside the PESAT application, namely

1. Encryption Module, for password encryption
2. Power Management Module, for shutdown, logoff and restart computer

There is only a user control inside the PESAT application, namely

1. Interface User Control, professionally written to manage all the button behavior

6.3.2 About PESAT

"About PESAT" Function is a brief introduction to the PESAT application, and contains the developer details as well as PESAT development version, initially the first version of 2004.

Implementation of Application Initialization Coding is as below:

```
Public Sub Init_My_Software() 'Initialize PESAT Application
On Error GoTo ErrHand
   Hive_Key = HKEY_LOCAL_MACHINE 'Registry HIVE Key
   Sub_Key = "Software\" & Software_Name 'Registry SUB Key
   If Open_SubKey(Hive_Key, Sub_Key) <> ERROR_SUCCESS Then
      blnInitial_Password = True 'Create Password
      Create_SubKey Hive_Key, Sub_Key, "Main" 'Path
      Open_SubKey Hive_Key, Sub_Key & "\Main"
      Create_Value REG_SZ, "Software Name", Software_Name 'Software Name
      Open_SubKey Hive_Key, Sub_Key & "\Main"
      Create_Value REG_SZ, "TPass", "TGFYWO[a" 'Initial Password as SECURITY
      Open_SubKey Hive_Key, Sub_Key & "\Main"
      Create_Value REG_SZ, "Software Path", WINDIR & "\PESAT.exe"
      Open_SubKey Hive_Key, Sub_Key & "\Main"
      Create_Value REG_SZ, "Installed Date", Date
      FileCopy App.Path & "\PESAT.exe", WINDIR & "\PESAT.exe" 'Enable in RUN
   End If
   Hive_Key = HKEY_LOCAL_MACHINE
   Sub_Key = "Software\Microsoft\Windows\CurrentVersion\App Paths"
   If Open_SubKey(Hive_Key, Sub_Key & "\PESAT.exe") <> ERROR_SUCCESS Then
      Create_SubKey Hive_Key, Sub_Key, "PESAT.exe"
      Open_SubKey Hive_Key, Sub_Key & "\PESAT.exe"
      Create_Value REG_SZ, "", WINDIR & "\PESAT.exe"
   Else
      Query_Value REG_SZ, ""
      If S_Value = "" Then
         Open_SubKey Hive_Key, Sub_Key & "\PESAT.exe"
         Create_Value REG_SZ, "", WINDIR & "\PESAT.exe"
      End If
   End If
Exit Sub
ErrHand:
   If Err.Number = 58 Then
      Kill (Desktop_DIR & "\Shortcut to PESAT.exe.lnk")
   Else
      MsgBox "Error Occurred while Initializing Software !", vbOKCancel, "PESAT Application"
   End If
End Sub
```

Implementation of Get System Directory Coding is as below:

```
Public Sub Get_System_DIRS() 'Get WINDOWS directory
On Error GoTo ErrHand
   Dim worked As Long
   WINDIR = String$(144, 0)
   worked = GetWindowsDirectory(WINDIR, Len(WINDIR))
   If worked = 0 Then
      MsgBox "Cannot Get WINDOWS Directory !!", vbOK, "PESAT Application"
      GoTo ErrHand
   Else
      WINDIR = Left(WINDIR, worked)
   End If
   WINSYSDIR = WINDIR & "\SYSTEM32" 'System Directory
   Hive_Key = HKEY_CURRENT_USER
   Sub_Key = "Software\Microsoft\Windows\CurrentVersion\Explorer\Shell Folders"
   Open_SubKey Hive_Key, Sub_Key
   Query_Value REG_SZ, "Start Menu"
   StartMenu_DIR = S_Value
   If Trim(StartMenu_DIR) = "" Then 'Start Menu
      StartMenu_DIR = WINDIR & "\Start Menu"
   End If
   Open_SubKey Hive_Key, Sub_Key
   Query_Value REG_SZ, "Desktop"
   Desktop_DIR = S_Value
   If Trim(Desktop_DIR) = "" Then
      Desktop_DIR = WINDIR & "\Desktop" 'Desktop
   End If
   Exit Sub
ErrHand:
   MsgBox "Error Occurred while Searching Windows Directory !" & "Software Cannot be
Loaded !", vbOK, "PESAT Application"
End Sub
```

Implementation of Form-Load Coding as below:

```
Private Sub Form_Load() 'Read All PESAT pre-setting
On Error Resume Next
   Get_System_DIRS 'Get Windows System Directory
   Get_All_Drives 'Get All Computer Drive
   Init_My_Software 'Initalize PESAT Registry
   Init_Password 'Password Management
   Read_AddRemove_Orphanage 'Read Add Remove Orphanage Page
   Read_Desktop_Page 'Read Desktop Page
   Read_Miscellaneous_Page 'Read Explorer Page
   Read_StartMenu_Page 'Read StartMenu Page
   Read_Aliases_Page 'Read Program Aliases Page
   Read_Restrict_Page 'Read Restrict Application Page
   fraPassword.Visible = False 'Frames is hidden
   fraRemoveOrphanage.Visible = False
   fraRegEdit.Visible = False
   fraDriveSec.Visible = False
   fraFolderSec.Visible = False
   fraAppRestriction.Visible = False
   fraAliases.Visible = False
   fraStartMenu.Visible = False
   fraDesktop.Visible = False
   fraVirusScanner.Visible = False
   fraMiscellaneous.Visible = False
   fraAbout.Visible = False
   fraPowerControl.Visible = False
   fraStart.Visible = True 'User Login Frame is visible
   cmdPassword.Enabled = False 'Command Buttons is hiden
   cmdOrphanage.Enabled = False
   cmdRegEdit.Enabled = False
   cmdDriveS.Enabled = False
   cmdFolderS.Enabled = False
   cmdRestriction.Enabled = False
   cmdAliases.Enabled = False
   cmdStartMenu.Enabled = False
   cmdDesktop.Enabled = False
   cmdVirusScanner.Enabled = False
   cmdMiscellaneous.Enabled = False
   cmdAbout.Enabled = False
   cmdPowerControl.Enabled = False
   No_Login = 0
End Sub
```

6.3.3 Password Management

"Password Management" Function is an interface for the user to change the preset password to their desire password. Initially, the password is set as SECURITY, with case sensitive. In order to change password, user has to first enter the old password correctly and enter twice the new password. If either any of the password is entered wrongly, the application decline the password changing request. When the application is launched, user would be asked to enter the login password. The other menu button would be hidden until the user has successfully login as a valid user. Three times of chances is given to the user to enter password. If there are three time wrong password in one launching, PESAT application quit automatically.

Implementation of Encryption - Decryption Coding is as below:

```
Public Function Cryption(strSource As String, strPassword As String, EnDeCrypt As Boolean) As
String
   'Encription And Decryption Class
   Dim intPassword As Long
   Dim intCrypt As Long
   For X = 1 To Len(strPassword)
      intPassword = intPassword + Asc(Mid$(strPassword, X, 1))
   Next X
   For X = 1 To Len(strSource)
      If EnDeCrypt = True Then
         intCrypt = Asc(Mid$(strSource, X, 1)) + intPassword + X
         Do Until intCrypt <= 255
            intCrypt = intCrypt - 255
         Loop
      Else
         intCrypt = Asc(Mid$(strSource, X, 1)) - intPassword - X

         Do Until intCrypt > 0
            intCrypt = intCrypt + 255
         Loop
      End If
      Cryption = Cryption & Chr(intCrypt)
   Next X
End Function
```

6.3.4 Remove Orphanage Entry

"Remove Orphanage" Function is a simple way to remove the junk entry in the Add Remove Programs in the Windows. Some of the entries in the Add Remove Programs are not being got removed correctly. Using this tweak, user can remove the orphanage entries easily.

6.3.5 Quick Access Panel

"Quick Access Panel" Function is a shortcut to many Windows API, namely

1. System Configuration Utility
2. Event Viewer
3. Registry Editor
4. Command Prompt
5. Control Panel
6. System Property
7. Display Property
8. Add Remove Program Wizard
9. Internet Property
10. Date and Time Property
11. Mouse Property
12. Sound and Multimedia Property
13. Regional Setting

Another enhancement in the PESAT application is giving the user an option to enable or disable the registry editor tool.

Implementation of Quick Access Panel Coding is as below:

```
Private Sub cmdSystem_Click() 'Launch System Property Applet
   Dim dblReturn As Double
   dblReturn = Shell("rundll32.exe shell32.dll,Control_RunDLL sysdm.cpl @1", 5)
End Sub

Private Sub cmdAddRemoveW_click() 'Launch Windows Add/Remove Programs Dialog
   Dim dblReturn As Double
   dblReturn = Shell("rundll32.exe shell32.dll,Control_RunDLL appwiz.cpl,,1", 5)
End Sub
```

```
Private Sub cmdDateTime_Click() 'Launch Windows Date/Time Properties Dialog
   Dim dblReturn As Double
   dblReturn = Shell("rundll32.exe shell32.dll,Control_RunDLL timedate.cpl", 5)
End Sub

Private Sub cmdDisplay_Click() 'Launch Windows Display Properties Dialog
   Dim dblReturn As Double
   dblReturn = Shell("rundll32.exe shell32.dll,Control_RunDLL desk.cpl,,0", 5)
End Sub

Private Sub cmdInternet_Click() 'Launch Windows Internet Properties Dialog
   Dim dblReturn As Double
   dblReturn = Shell("rundll32.exe shell32.dll,Control_RunDLL inetcpl.cpl,,0", 5)
End Sub

Private Sub cmdMouse_Click() 'Launch Windows Mouse Properties Dialog
   Dim dblReturn As Double
   dblReturn = Shell("rundll32.exe shell32.dll,Control_RunDLL main.cpl @0", 5)
End Sub

Private Sub cmdMultimedia_Click() 'Launch Windows Multimedia Properties Dialog
   Dim dblReturn As Double
   dblReturn = Shell("rundll32.exe shell32.dll,Control_RunDLL mmsys.cpl,,0", 5)
End Sub

Private Sub cmdRegional_Click() 'Launch Windows Regional Settings Dialog
   Dim dblReturn As Double
   dblReturn = Shell("rundll32.exe shell32.dll,Control_RunDLL intl.cpl,,0", 5)
End Sub
```

6.3.6 Drive Security

"Drive Security" Function automatically detect all the drive that exists in the computer, listing down in a check-list box, so that the user has the right to hide any listed drive. Supported drives with this function are floppy drive, single-partitioned hard drive, multiple-partitioned hard drive, CD-ROM, CD-RW and DVD.

Implementation of Drive Security Function is as below:

```
Private Sub Apply_Hide_Drives() 'Drive Security Setting - Hide Drive
On Error GoTo ErrHand
   Dim lTot_Hide_Number As Long
   Dim i As Integer
   lTot_Hide_Number = 0
   For i = 0 To lstDrive.ListCount - 1
      If lstDrive.Selected(i) = False Then
         lTot_Hide_Number = lTot_Hide_Number + _
          get_Hide_Drive_Number(Left(lstDrive.List(i), 1))
      End If
   Next
   Hive_Key = HKEY_CURRENT_USER
   Sub_Key = "Software\Microsoft\Windows\CurrentVersion\Policies\Explorer"
   Re_Init_Val
   If lTot_Hide_Number <> 0 Then
      Open_SubKey Hive_Key, Sub_Key
      Create_Value REG_DWORD, "NoDrives", lTot_Hide_Number
   Else
      Open_SubKey Hive_Key, Sub_Key
      Delete_Value "NoDrives"
   End If
Exit Sub
ErrHand:
   MsgBox "Error Occurred while Hiding Drives !", vbOK, "PESAT Application"
End Sub
```

6.3.7 Folder Security

"Folder Security" Function is able to lock the selected folder firmly. The interesting part of this function is the lock-folder is changed to other icon and the path of the source folder does no longer exist. And the result, the folder would be protected all round. User can select Recycle Bin Icon, My Computer Icon, Control Panel Icon and Printer Icon as their destination for the protected folder.

Implementation of Folder Security Coding is as below:

```
Private Sub cmdSecure_Click() 'Secure the selected folder
On Error GoTo ErrHand
   Dim Path, Data, File, Ext, FileName As String
   If Option1.Value = True Then 'My Computer Icon
      Ext = ".{20D04FE0-3AEA-1069-A2D8-08002B30309D}"
      GoTo Excecute
   ElseIf Option2.Value = True Then 'Control Panel Icon
      Ext = ".{21EC2020-3AEA-1069-A2DD-08002B30309D}"
      GoTo Excecute
   ElseIf Option3.Value = True Then 'Printer Icon
      Ext = ".{2227A280-3AEA-1069-A2DE-08002B30309D}"
      GoTo Excecute
   ElseIf Option4.Value = True Then 'Recycle Bin Icon
      Ext = ".{645FF040-5081-101B-9F08-00AA002F954E}"
      GoTo Excecute
   End If
Excecute:
   Path = dirDir.Path
   Data = Mid$(Path, InStrRev(Path, "\") + 1, Len(Path))
   File = Left$(Path, Len(Path) - Len(Data))
   If Not UCase$(Path) = UCase$(WindowsDirectory) _
   And Not UCase$(Data) = UCase("desktop") Then
      FileName = File & Data & Ext
      Name dirDir.Path As FileName
      dirDir.Path = File
      MsgBox "Folder Locked Successfully!", vbOK, "PESAT Application"
   Else
      MsgBox "Folder Error While Locked...Please double click the folder!", vbOK, "PESAT
Application"
   End If
Exit Sub
ErrHand:
   MsgBox "Folder Error While Locked...Please double click the folder!", vbOK, "PESAT
Application"
End Sub

Private Sub cmdUnsecure_Click() 'Unsecure the selected folder
On Error GoTo ErrHand
   Dim Path, temp, Data, File, Ext, FileName As String
   Path = dirDir.Path
```

```
    temp = Mid$(Path, InStrRev(Path, "\") + 1, Len(Path))
    Data = Left$(temp, InStr(temp, ".{") - 1)
    File = Left$(Path, Len(Path) - Len(temp))
    FileName = File & Data
    Name dirDir.Path As FileName
    dirDir.Path = File
    MsgBox "Folder Unlocked Successfully!", vbApplicationModal + vbInformation, "PESAT
Application"
    cmdApply.Enabled = True
    Exit Sub
ErrHand:
    MsgBox "Folder Error While Unlocked...Please double click the folder!", vbOK, "PESAT
Application"
End Sub
```

6.3.8 Application Restriction

"Application Restriction" Function paves the user the option to restrict all the application in the computer from running, except for those that are permitted in the Allow List. This is a very strong function that really turns down the entire restricted program from running. User can only turn down the restriction on the program through the PESAT interface. Allow List in this application is a list for the user to add or delete the desire program from running.

6.3.9 Application Aliases

"Application Aliases" Function list down the available program in the computer and the user just have to type in the alias name in the RUN menu to launch the application. User is also able to create a new alias, for just only put a name and locate it to the correct path. Then, type the name in the RUN menu, the program would launch expectedly.

6.3.10 Start Menu Tweak

"Start Menu Tweak" Function provides simple tweak to personalize the StartMenu. It contains the numerous options for user to tweak the StartMenu as below.

1. Remove the "Run" command form the StartMenu
2. Remove the "Favorites" command from the StartMenu
3. Remove the "Search" command from the StartMenu

4. Remove the "Documents" command the StartMenu
5. Add an expanding Control Panel to the StartMenu
6. MenuShowDelay, by default is set to 400

6.3.11 Desktop Tweak

"Desktop Tweak" Function provides simple tweak to manage some of the desktop customization as below.

1. Hide all Desktop Icon
2. Hide the Internet Explorer Icon
3. Disable the right-click menu on Desktop
4. Clear the RUN command history

6.3.12 Miscellaneous Tweak

"Miscellaneous Tweak" Function is a simple tweak to approach many hidden function in the Windows XP.

1. Add a menu option "Copy Folder" to every folder
2. Add a menu option "Move Folder" to every folder
3. Set a BitMap, bmp on the Internet Explorer toolbar

Implementation of Miscellaneous Tweak Coding is as below:

```
Public Sub Read_Miscellaneous_Page() 'Miscellaneous Tweak
On Error GoTo ErrHand
   Hive_Key = HKEY_CURRENT_USER
   Sub_Key = "Software\Microsoft\Windows\CurrentVersion\Explorer"
   'Display Background Bitmap on Explorer Toolbars
   Hive_Key = HKEY_CURRENT_USER
   Sub_Key = "Software\Microsoft\Internet Explorer\Toolbar"
   Open_SubKey Hive_Key, Sub_Key
   If Query_Value(REG_SZ, "BackBitmapShell") = ERROR_SUCCESS Then
      If S_Value <> "" Then
         If LCase$(Right(S_Value, 4)) = ".bmp" Then
            frmMain.chkDisplayBGImage.Value = 1
            frmMain.txtBGImage = S_Value
         Else
            frmMain.chkDisplayBGImage.Value = 0
         End If
      Else
         frmMain.chkDisplayBGImage.Value = 0
      End If
```

```
    Else
        frmMain.chkDisplayBGImage.Value = 0
    End If
    Hive_Key = HKEY_CLASSES_ROOT 'Add a Menu Option to Copy Folders
    Sub_Key = "Directory\shellex\ContextMenuHandlers\Copy to Folder"
    If Open_SubKey(Hive_Key, Sub_Key) = ERROR_SUCCESS Then
        Query_Value REG_SZ, ""
        If S_Value = "{C2FBB630-2971-11d1-A18C-00C04FD75D13}" Then
            frmMain.chkAddCopyFolder.Value = 1
        Else
            frmMain.chkAddCopyFolder.Value = 0
        End If
    Else
        frmMain.chkAddCopyFolder.Value = 0
    End If
    Hive_Key = HKEY_CLASSES_ROOT 'Add a Menu Option to Move Folders
    Sub_Key = "Directory\shellex\ContextMenuHandlers\Move to Folder"

    If Open_SubKey(Hive_Key, Sub_Key) = ERROR_SUCCESS Then
        Query_Value REG_SZ, ""
        If S_Value = "{C2FBB631-2971-11d1-A18C-00C04FD75D13}" Then
            frmMain.chkAddMoveFolder.Value = 1
        Else
            frmMain.chkAddMoveFolder.Value = 0
        End If
    Else
        frmMain.chkAddMoveFolder.Value = 0
    End If
Exit Sub
ErrHand:
    MsgBox "Error Occurred while Reading Explorer Settings !", vbOK, "PESAT Application"
End Sub
```

6.3.13 Virus Scanner Tweak

"Virus Scanner Tweak" Function will show the very last time the Norton Antivirus 2003 update its virus definition and the virus definition package date. By then, user can launch the Norton LiveUpdate immediately for automatically download, or browse the Symantec Norton Antivirus Definition Download Page to select other virus definition package for manual download.

Another special feature of PESAT application is it contains an EICAR string test that can simulate a virus in the computer to test whether the Antivirus Application can work efficiently. It would prompt a message after the creation of the virus, then on the normal

condition; Antivirus would come to an alert message for the virus detection and deletion. If not, the Antivirus Application software inside the computer might be mal-function.

Implementation of EICAR Coding is as below:

```
Private Sub cmdTrigger_Click() 'Create Virus Text String
On Error GoTo ErrHand
Dim OP As Integer
Dim VRSTR As String
   VRSTR = "X5O!P%@AP[4\PZX54(P^)7CC)7}"
   VRSTR = VRSTR + "$EICAR-STANDARD-ANTIVIRUS-TEST-FILE!$H+H*" 'EICAR String
   OP = 0
   Open "C:\VIRUSTEST.TXT" For Output As 1
   Print #1, VRSTR
   txtVirus.Text = "Virus Sring has been created"
   Close 1
   OP = 1
   cmdApply.Enabled = True
   Exit Sub
ErrHand:
   MsgBox "Error Occurred while Adding Program Alias !", vbOK, "PESAT Application"
End Sub
```

6.3.14 Power Management

"Power Management" Function can LogOff, ShutDown and Restart the computer immediately. This is the very fast way to executer the power management command. After the confirmation, all the application would be terminated immediately, and the computer will either in the state of LogOff, ShutDown or Restart instantly.

Implementation of Power Management Coding is as below:

```
Private Sub butExecute_Click() 'Power Control for Logoff, Restart, Shutdown
On Error GoTo ErrHand
   Dim sPower As New clsShutdown 'clsShutdown in Class Modules
   If optLogOff.Value = True Then 'Logoff
      MsgBox "Windows is now Log Off...", vbOK, "PESAT Application"
      sPower.ExitWindows WE_LOGOFF
      DoEvents
   ElseIf optRestart.Value = True Then 'Restart
      MsgBox "Windows is now Restart...", vbOK, "PESAT Application"
      sPower.ExitWindows WE_REBOOT
      DoEvents
   ElseIf optShutDown.Value = True Then 'Shutdown
      MsgBox "Windows is now Shut Down...", vbOK, "PESAT Application"
      sPower.ExitWindows WE_SHUTDOWN
      DoEvents
   End If
   Exit Sub
ErrHand:
```

```
    MsgBox "Error Occurred while running Windows Power Management!!", vbOK, "PESAT
Application"
End Sub
```

6.4 System Documentation

System Documentation is an important process in the process development. Consequently, detail system documentation is well prepared for the PESAT application. See Appendix B for more details. System Documentation is a reference or guide to the user. It explains and describes how the PESAT application can be used in the most proper way. It reduces the learning curves of the user and save their time.

6.5 Summary

The development of PESAT application has been carefully planned and it does take a period of precious time. A lot of the functionalities as described in the previous chapter have been implemented orderly in time. Some functions that are not fully implemented would be extended in future. In short, PESAT application has fully utilized the Microsoft Technologies to build up the system. Whether the application is strong enough to be the security tool depends on the testing. The next chapter would draw the system testing procedures on PESAT application.

CHAPTER 7
SYSTEM TESTING

7.1 Introduction to System Testing

System testing is a significant and critical phase that ensures the system fulfills the user's requirements and assures the quality of the delivered system.

Testing provides a method to discover logical error and to test the system reliability. It is done throughout system development, not just at the end. This is because system that is failed after installation will result a waste in cost, time and effort. However successful testing will result in quality software with less errors and work according to specification.

Several testing stages that involve during the development of the system are:

1. Unit testing
2. Integration testing
3. System testing

7.2 Unit Testing

In this stage, testing will be concentrated on the smallest component of the system for testing. Each individual component is tested independently without other system

components, to ensure that they operate correctly. For example, this component might perform task like checking valid input value.

In PESAT application, some of the units that were tested independently are:

1. The password is encrypted correctly.
2. The password is kept in PESAT registry.
3. There are PESAT registry group.
4. All SHELL command can link to the Window API.
5. Drive can be hidden successfully.
6. Folder is securely locked.
7. There is no application to be launched if PESAT is in the state of "Restrict All Application".
8. All tweak in StartMenu, Desktop and Miscellaneous can work independently and successfully.
9. Virus Scanner can correlate with the Norton Antivirus 2003 successfully.
10. One key to LogOff, Restart and ShutDown Computer immediately.

PSEAT application has integrating three types of unit testing method to test each of the unit listed above, namely

1. Adhoc Testing
2. White Box Testing
3. Black Box Testing

7.2.1 Ad Hoc Testing

Ad Hoc or ad lib testing means simply play with the functioning unit, trying whatever comes to their mind, in attempt to make it fail. This type of testing was a fast and efficient way of debugging code errors during the early development stage.

The disadvantage of Ad Hoc testing is it usually finds many errors and never be sure what was or was not to be tested.

7.2.2 White Box Testing

White Box Testing basically involved analyzes the structure of the code and use knowledge about the structure of a component to derive test data. The advantage of white box testing is that an analysis of the code can be used to find out how many test cases are needed to guarantee a given level of test coverage.

Many codex testing is conducted using the white box testing approach. This includes the basic testing on path like Windows Directory, data flow and loop testing. In short, the white box testing methodology is focused in term of coverage. And the vital part of this testing is to check for any missing function.

7.2.3 Black Box Testing

Black Box Testing is concentrate on the functionality of code. The main objective is to uncover those wrong functions programmed correctly by feeding the input to the black box and take notes on what output is produced. The test object's behavior can only be determined by studying its inputs and the related outputs.

The advantage of this kind of testing is that a black box is free of the constraints imposed by the internal structure and logic of the test object. However the disadvantage is that it is not always possible to run a complete test in this manner. Black box testing is a test on boundary value analysis, error guessing and domain testing.

7.3 Integration Testing

After all components have been unit-tested, the next step is ensuring that the interfaces among the components are defined and handled properly. This step is called integration testing, also known as module testing, which verifies that the all the components work together as described in the module or system design specifications.

During the integration testing, two or more units in which either unit that use output data from or provide input data for another unit were tested in collection. The combination of many modules would test the integration of the system. Some integration may result an unexpected error. Using the integration testing approach, this kind of errors will be discovered and corrected.

The order in which components are tested affects our choice of test cases and tools. The system is viewed as a hierarchy of components, where each component belongs to a layer of the design. In this system, the Top-down Integration approach is used where testing begins from the top and works the way down. The process is continued until all the modules are tested.

7.4 System Testing

System Testing is the last testing procedure. It is performed to uncover its limitations, measure it capabilities and make certain that the entire system works according to users' specifications. Developers will join the users to perform this stage of testing where the system is checked against the users' requirements description.

System modification will be implemented if there is a need to change or do not met the users' requirements specifications. If the users are satisfied with the system's characteristics, the system is ready to be deployed for use. The testing result will show whether or not the entire system specifications and objectives are achieved.

7.4.1 System Test Consideration

In system testing, the behavior of the individual functions and functional tests also involved:

1. The Event List
2. Error Message Testing
3. Security Testing
4. Transaction Testing

The Event List

All the possible triggers are exercised and the expected results compared with the actual results. Every function is tested by one or more events in the event lists.

Error Message Testing

The error message, which can be generated by the system during invalid data entry are checked for spelling, appropriateness and consistence. Acknowledgement messages also will also implement the same test. It is the message that informs the user about the state of a user request process. For instance, if there is no Norton Antivirus 2003 in the computer, PESAT application would result an alert message to notify the user about it. The overall of the result was satisfactory although some modification had to make.

Security testing

In security testing, the system is tested for improper penetration and unauthorized access, to ensure that the implementation of the user login and the valid user checking procedures included in every authorized page are functioning accordingly and correctly. The test had show that the security function is working properly.

Transaction Tracking

During transaction tracking, a list of possible transactions is tracked through the system to ascertain that they function correctly from input to output. For example, every time a screen is reached which requires input or generates input, the appropriate functions are processed and lead to subsystem for processing and then the right output is retrieve. This test was implemented and all the function behaves according to the requirements specification.

7.5 System Fundamental Testing

There are other tests fundamental to all software. Certain of these are difficult to measure accurately. Five of these fundamental tests are:

1. Usability
2. Installability
3. Performance
4. Reliability

Usability

The usability should be based in building user interfaces that have patterns already familiar to the typical user. The user then learns to use the software through pattern matching and paradigm shifts, exactly as they do in mastering any product. It is an expectation to PESAT application to be very user-friendly so that the user can drive the application smoothly and efficiently.

Install Ability

How easy is it for a novice to install the software correctly and easily without recourse to an expert? This is the concept of the Installability. PESAT application is expected to be installability to most of the user with basic knowledge in computer. It is a typical installation of software setup that would guide user through out the process.

Performance

Performance test are conducted to ensure that the system response time meet user expectations and does not exceed the specified performance criteria under heavy stress or volume. During these tests, response time and the transaction rate are measured; the purpose of performance tests is to test-run the performance of various functions of the software within a specified hardware configuration. The performance tests can couple this test with stress testing.

Reliability

Reliability tests are conducted, according to mathematical models of software reliability, to ensure that the system can be probability of some function of the system failing within a specified time. Reliability testing is monitoring the mean time between failures. Reliability and consistency testing go hand in hand where the system behavior (inputs, outputs, response time) is measured for consistency. PESAT application must be reliable to deny the access of unauthorized user so that it is secure enough for a valid computer user to protect the sensitive data which reside in the computer.

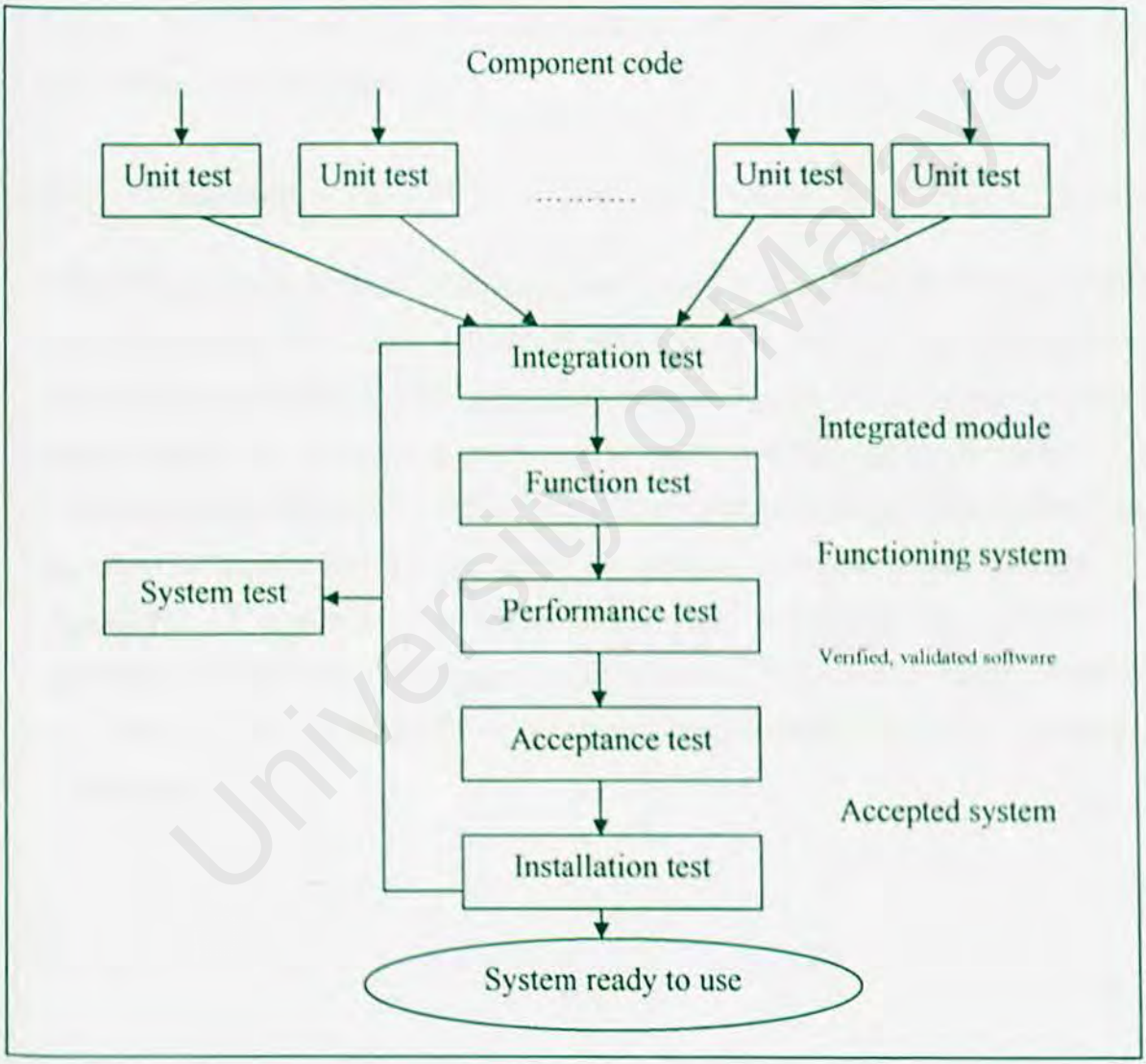


Figure 7.1 Testing Steps

7.6 System Maintenance

Usually in system development, maintenace will be conducted once the system is finished or delivered. The maintenace servises would assure the system function can perform proporly and free of bugs. It also includes the activities to modify some of the application or add new functions into the system. For the time being, PESAT application is on the state of development and the system maintenance is not suited in this stage. PESAT application would undergo the maintainance service once it is depoyed in the computer system. From there, it is predictable that PESAT application will extend many more functions in the future.

7.7 Summary

PESAT application has been undergone many testing to assure that the entire application is strong enough and no defect when user drive it in their computer. As a security and administration software, PESAT application must have security measurement so that those unauthorized users would not be able to intrude into the application easily. Another point to be noticed is PESAT application as an administrative application must give the user an easy way to tweak any of the setting. Consequently, all functions beyond PESAT application must be carefully and fully tested in all ways so that the application would likely to be the perfect application. The following chapter would deals the system evaluation on PESAT application and the conclusion on PESAT application development.

CHAPTER 8

SYSTEM EVALUATION AND CONCLUSION

8.1 Introduction to System Evaluation And Conclusion

Evaluation is a process that occurs continuously at all phases of the system development. Evaluation phase was to determine the extent to which the system the expected outcomes have been realized, and the prescriptive value of the process where extraneous factors were taken consideration. Lastly, conclusion will be making for this system.

8.2 System Evaluation

System Evaluation in the PESAT application contains 4 different elements to be discussed. The coverage of the evaluation as below:

1. Difficulty and Suggested Solution
2. System Strength
3. System Limitation
4. Future Enhancement

8.2.1 Difficulty And Suggested Solution

In every project, the problems always occur during system development. This does not vary to PESAT application development. Many problems have kept unfolding one after another as development work progressed due to many reasons. Some of the problems are

unexpected and difficult to be overcome. Many of the problems can be handles rather easily.

During Analysis Phase

- *Determining Scope of the PESAT application*

Developing a programmatic system does require a lot of experiences. During the PESAT application development, it is very hard to determine to what extend the scope of the application, due to the lack of experiences. The scope of the application must take consideration on time constraint so that it can be completed within the period of time. However, this is handled right after studying and analyzing several applications and their capabilities with the same functionalities as PESAT application suggested. Then the whole picture of PESAT application is cleared and the scope is determined on the basis of moderately.

- *Programming Languages*

PESAT application can be developed with many programming languages. For instance, there are Java, Pearl or Visual C++ programming technologies. It is very hard to decide which language to be used initially, because every programming language has their strength. After studying and analyzing the details of each of the programming languages, the fact is JAVA is not very well suit for Windows API , Pearl is too old programming language, Visual C++ is hard to program the interface, and finally, Visual Basic is concluded to be the PESAT application core programming language. It is due to Visual Basic is absolutely a language to program API and user-friendly-interface.

During Design Phase

- ***Time Constraint***

There was not enough time to study, learn and produce the best solution of design in PESAT application. Mainly, this was cause by inexperience and insufficient knowledge of designing a full extend of an application software. Furthermore, time is needed to study and explore Windows XP and the programming languages deeply before knowing how to integrate these technologies and languages in the process of developing and

solving problems. Thus the best way is to study as many approaches used in senior and previous year students' documentation.

During Implementation Phase

- *No prior experience in the chosen programming languages*

There was a learning curve in understanding how the Visual Basic and Windows Script Host works since inexperience in those languages. Scripting in a new environment such as WSH requires some knowledge of what the API objects do and how to use the objects to build the required functionality of the PESAT application. The best way of learning the coding is to study as many examples as possible. Another way to studying the programming coding is to refer the references book.

During Testing Phases

- *Problem on testing*

Most of the tweaking in PESAT application needs to restart the computer so that the setting would take effect. This is rather a time wasting to wait the computer to restart, especially in the situation of lack of time. There is no other shortcut way to accelerate the testing phase. Consequently, the testing phase is conducted as early as possible. Using the Rapid Application Development, RAD approach, most of the functions in the PESAT application is tested in the time of development as all the functions in PESAT application can be tested independently.

8.2.2 System Strength

PESAT application has many strengths compare to the current similar application in the market. This can be well-described as follow.

System Transparency

PESAT application has a high level of system transparency. In term of system transparency, the user does not need to know where the PESAT registry resides, how the application is structured and how to edit the registry value in the Windows registry. The

most importantly, the user only need to know how to communicate with the PESAT application; and the PESAT application will communicate with the Windows system.

Systematic Error Handling

Input of the users will be validated and verified to prevent the errors caused by invalid input. If there is any error or invalid input occurred, an error message is generated and displayed to inform the user about the error. For example re-enter username and password will be prompt out when users input the invalid username or password.

Registry Conceptual

PESAT application is using the registry in the Windows to manipulate most of the functionalities. Some of the Windows registry is giving value, re-value, or even creating a new registry to store the data. This is a brilliant way to keep data within the Windows, and the benefit, it does not consume much data.

Outstanding Interface

PESAT application has adapted a brand new user-friendly interface. It is an outclass work of design. It has ocean blue as the background, interlace with many floating silver-shown button and white in frames. A conclusion to the interface is the PESAT application is really user-friendly, nicely organized, elegantly arranged and sooth to the eyes as well.

Perfect Windows Tweaking

PESAT application has been designed to let the user tweak the Windows operating system as they like. Most of the users do not know to configure their operating system to fit their need. Initially, PESAT application offers part of the registry tweaking for the user's convenience. Ranging from the StartMenu tweak, PESAT application can tweak the Desktop, Toolbar, and RightMouseMenu and many more exciting tweaking.

Uncover Windows Functionality

Some of the tweaking in the PESAT application dealing to the registry is the hidden function in the Windows operating system. For instance, the drive visibility and StartMenuDelay. PESAT application has uncovered the hidden Windows tweaking to the user. Using this application, user is able to approach the Windows operating system in depth, and most importantly, easily.

Security Application

PESAT application is not only the administration application, but also the security console application. One of the most outstanding functions in the PESAT application is the Folder Locker. It can lock any of the folders that contain sensitive information easily. The user can only open the locked folder through the PESAT application. PESAT application can also control the drive visibility, conduct password encryption and even restrict the usage of the installed program.

Antivirus Application Integration

PESAT application is initially integrating the Norton Antivirus 2003 to the system. This means that user can use Norton Antivirus through the PESAT application. The integration is expected to be extended with other antivirus application in future.

8.2.3 System Limitation

From the application strength mentioned as above, PESAT application do has some of the limitation. This is expected as PESAT application is only the first edition application. In addition, the limitations due to the time constraints are not avoidable. This is described as below.

Multiple User Account Operating System

PESAT application is not fully correct the problem in the multiple user account operating system. In the multiple user account operating system, the security features in the application can be turned down. This is due to the PESAT application tweaking is only effective to the user account where it is installed. Thus, PESAT application does

not support multiple user account in the computer. It suit only for personal-use computer.

Operating System

PESAT application is developed for the need in the Windows XP. In the operating system other than that, PESAT application might not fully functional at all. Different operating system has different system files, and the result, PESAT application can not recognize all of the operating system. It is best fit in the Windows XP.

Antivirus Application

For the time being, PESAT application can only works in pair with Norton Antivirus 2003 in the Intelli Virus Scanner Module. It can not stand with other virus scanner. The limitation is reasonable as PESAT application is initially using Norton Antivirus as the application prototype and is expected to extend the Virus Scanner Function to feed most of the major Antivirus Application.

Languages

PESAT application at the present time only supports English version language. It does not translate to other language like Malay or Chinese.

Performance Issue

At the present time, for some of the function, PESAT application only takes effect right after the computer reboot. The fact is that the situation would consume some of the user time. In such, PESAT application performance is decreased a little.

8.2.4 Future Enhancement

PESAT application at the present time does have many spaces to be evolved and extended to cover many more exiting functions. It is a customizable application to feed the need of the contemporary user. A lot of the administration function and security function can be included in the PESAT application in the future. Here are some of the future enhancements that is expected.

Application Start-Up

PESAT application should be start-up automatically upon the Windows online. By then, it must be placed in the taskbar so that the user can easily launch PESAT application.

Password Recovery

PESAT application can include this Password Recovery function in future. In such, Password Recovery can unmask the mask password so that the user can retrieve and recover their password if they forget the password.

Service Manager

PESAT application can include the Service Manager so that the user can easily know what is running in their operating system background.

Port Visibility

PESAT application should include the Port Visibility function so that the user can know which port is opened at the present time and sense the unknown listen-port as it might be intruded by unauthorized users.

Internet Management

PESAT application can attach the Internet Management as part of the tweaking function. In term of Internet Management, PESAT application can clear the internet cookies, history file, typed-URL and many other functions relate to Internet.

Operating System

PESAT application would be extended to all the Windows Family Operating System like Windows Server 2003, ME, 2000 and 98. If so, all sort of operating system would be able to use PESAT application in their operating system, no matter which version of operating system they used.

Virus Scanner

PESAT application would extend the Intelli Virus Scanner in the application so that it an works with other virus scanner like MacAfee, Sophors, PC-Cilin and other Antivirus Scanner.

Language

PESAT application should have a Malay version so that people of Malaysian can have their own application in the native language.

Multiple User Account Operating System

Current PESAT application is only deployed in the personal computer. In future, PESAT application is expected to be used in the network environment. If so, PESAT application should be developed to support multiple user account operating system.

8.3 Project Conclusion

PESAT application is designed specifically as to be a bundle of software that merges the security element and administration element into a software package. This is an idea from the market reviewing that the contemporary software bazaar is lacked of this sort of application. Some applications has the security functions but lack of administration part; while the vice verse, some contain administration functions but without the security part. Consequently, PESAT application is the outcome from the weaknesses in the contemporary software bazaar.

In the development of PESAT application, many measurements has been taken to ensure that the outcome of PESAT application does suit the need of many computer users, no matter the novice or the professional user. It is highly created on the basis of user-friendly interface, reliable and securable systems do not consume many computer resources. The result, PESAT application has meet all the quality stated above.

The development of PESAT application is using the Rapid Application Development methodology. This is a very excellent methodology to develop the application with

many functions, as one by one of the function can be tested upon the completion of each module. The benefit of this methodology is PESAT application does not has to wait until all of the application to be built up completely for the testing phase. This accelerates the development of the PESAT application.

Throughout the development of PESAT application, a lot of the programming skills are required. This enhances and empowers the developer logical yet critical thinking of the developer to polish the programmatic skills, especially in Windows API Programming. From the PESAT application, it is noticeable that Windows Operating System contains a lot of the hidden functions that a novice user would not be able to tweak the setting. By then PESAT application approaches the novice user to explore all sort of Windows tweaking in their operating system and instill a fact to the user that the operating system is customizable to feed every of their need.

In is very high expectation that PESAT application would be extended the functionality in the future. Some tweaking like Internet Management, Port Visibility, Zip-File Recovery and other useful tweaking should be included in the future version of PESAT application. As a conclusion, PESAT application is indeed excellent software to be tried and used in every computer user.

REFERENCE

[1] H.M. Deitel., P.J. Deitel., Visual Basic. Net, How To Programming, 2nd Edition, Prentice Hall, 1999, United States of America.

[2] H.M. Deitel., P.J. Deitel., Visual Basic 6, How To Programming, 3rd Edition, Prentice Hall, 1999, United States of America.

[3] Kendall, K.E. and Kendall, J.E., System Analysis and Design, 4th Edition, Prentice Hall, 1999, United States of America.

[4] Charles P.Pflexxger and Shari Lawrence Pfleeger, Security in Computing, 3rd edition, Pearson Educational International.

[5] Halting the Hacker, A Practical Guide to Computer Security, 2nd Edition, Hewlett-Packard Professional Books Series.

[6] Eric Maiwald, Network Security, A Beginner Guide, Osborne Publication, 1st Edition.

[7] System Administration Scripting Guide Script Repository E-Book, by Microsoft Publication.

[8] http://www.cert.org/tech_tips/home_networks.html

[9] http://www.utoronto.ca/security/pcsec.htm

[10] http://nsi.org/Library/Compsec/faq.htm

[11] http://nsi.org/Library/Compsec/goodpass.html

[12] http://www.utoronto.ca/security/virus.html

[13] http://www.setsolutions.com/security.htm

[14] http://support.discusware.com/center/resources/admdoc40/started.php

[15] http://www.ncsa.uiuc.edu/UserInfo/Resources/Hardware/IBMp690/IBM/usr/share/man/info/en_US/a_doc_lib/aixbman/admnconc/sys_mgmt.htm

[16] http://www.tropsoft.com/

[17] http://www.microsoft.com/

[18] http://www.glocksoft.com/amlv/index.htm?source=AAToolsMoreProd

[19] http://www.totalidea.de

[20] http://www.karchitects.com/

[21] http://www.giant-technologies.com/hyena/

[22] http://www.softheap.com/secagent.html

[23] http://www.reliablesoftware.com/articles/A%20Programmer's%20Eye-View%20of%20.NET%20Security%20Administration.htm

[24] https://www.microsoft.com/technet/scriptcenter/scrguide/sas_wmi_tjin.asp?frame=true

[25] http://www.credata.com/research/methodology.html

[26] http://sysdev.ucdavis.edu/WEBADM/document/rad-stages.htm

[27] Microsoft MSDN Library Visual Studio 6

[28] Microsoft Visual Studio. Net Documentation

[29] JAVA 1.4 SDK Documentation